BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images,
Please do not report the images to the
Image Problem Mailbox.

Parallel pseudo-random generator for emulating a serial pseudo-random generator and method for carrying out same.

Patent Number:	EP0397079, A3
Publication date:	1990-11-14
Inventor(s):	ROGER GEORGE ANDRE CHARLES (FR); POWELL WILLIAM EDWARD (US); WEEBER WILLIAM BERNARD (US)
Applicant(s):	ALCATEL NV (NL)
Requested Patent:	JP3068022
Application Number:	EP19900108555 19900507
Priority Number (s):	US19890351175 19890512
IPC Classification:	H04L25/49
EC Classification:	H04L25/03E3
Equivalents:	AU5495690, AU629933, CA2016635, US5031129
Cited patent(s):	US4965881

Abstract

A parallel pseudo-random generator for emulating a serial pseudo-random generator that generates serial outputs such that the next serial output value is based upon an Exclusive OR combination of at least two preceding serial output values the maximum preceding serial output value defined as the Pth preceeding serial output value, where P is an integer greater than one; comprising: A) at least P latches, each latch having an output having a logic value 1 or 0 and an input operable upon receipt of a clock signal, for receipt of data for controlling the next logic value on the latch output; B) at least P Exclusive OR gates, each having at least two inputs and one output, each Exclusive OR gate output connected to a corresponding input of one latch so as to define the next value of the latch output upon receipt of the next clock signal; and C) means for connecting each input of each Exclusive OR gate to one latch output so that the output of each Exclusive OR gate represents the corresponding next value of the latch to which This Exclusive Or gate output is connected.

Citation 4

⑲日本国特許庁(JP)　⑪特許出願公開

⑫公開特許公報(A)　平3-68022

⑤Int. Cl.⁵	識別記号	庁内整理番号	㊸公開　平成3年(1991)3月25日
G 06 F 7/58	B	7056-5B	
H 03 K 3/84	A	8626-5J	

審査請求　未請求　請求項の数　4　(全31頁)

㊹発明の名称　直列疑似乱数列生成機をエミュレートするための並列疑似乱数列生成機及びその実行方法

㉑特　願　平2-122861
㉒出　願　平2(1990)5月11日
優先権主張　㉜1989年5月12日㉝米国(US)㉛351,175

㉗発明者　ウイリアム・エドワード・ボーウエル　アメリカ合衆国、ノース・カロライナ・27614、ローリ、トロッターズ・リッジ・ドライブ・201
㉗発明者　ウイリアム・バーナード・ウエバー　アメリカ合衆国、ノース・カロライナ・27502、アペックス、ブラニー・フランクス・ロード・7917
㉑出願人　アルカテル・エヌ・ベー　オランダ国、1077・イツクス・イツクス・アムステルダム、ストラビンスキーラーン・341
㉔代理人　弁理士　川口　義雄　外2名
最終頁に続く

明　細　書

1．発明の名称

直列疑似乱数列生成機をエミュレートするための並列疑似乱数列生成機及びその実行方法

2．特許請求の範囲

(1)　次の直列出力値が少なくとも2つの先行の直列出力値の排他的ORの組合せに基づくような直列出力を生成する直列疑似乱数列生成機をエミュレートするための並列疑似乱数列生成機であって、最大の先行直列出力値がP番目(但しPは1よりも大きい整数である)の先行の直列出力値として定義されており、

A) 各々が、論理値1または0を有する出力と、前記出力において次の論理値を制御するためにデータを受信するようにクロック信号受信時に動作可能な入力とを有する少なくともP個のラッチと、

B) 各々が少なくとも2つの入力と1つの出力とを有する少なくともP個の排他的ORゲートであって、該排他的ORゲートの各々の出力が、次のクロック信号受信時に前記ラッチの出力の次の値を定義するように、1つのラッチの対応する入力に接続されている排他的ORゲートと、

C) 前記排他的ORゲートの各々の出力が、この排他的ORゲートの出力が接続されているラッチの対応する次の値を与えるように、前記排他的ORゲートの各々の入力を1つのラッチの出力に接続する手段

とを包含する並列疑似乱数列生成機。

(2)　前記直列疑似乱数列生成機を定義する直列排他的OR組合せが、(Pを7とすると)6番目及び7番目の先行の直列出力値に基づいてその次の出力値を決定する請求項1に記載の並列疑似乱数列生成機。

(3) 前記ラッチの数が8であり、該ラッチが対応する出力Q0からQ7を有しており、各々が対応するラッチの入力に接続されている出力を有する対応する排他的ORゲートEx0からEx7の入力が、

Ex0の入力がQ4及びQ6に接続され、

Ex1の入力がQ5及びQ7に接続され、

Ex2の入力がQ0及びQ1に接続され、

Ex3の入力がQ1及びQ2に接続され、

Ex4の入力がQ2及びQ3に接続され、

Ex5の入力がQ3及びQ4に接続され、

Ex6の入力がQ4及びQ5に接続され、

Ex7の入力がQ5及びQ6に接続される

ようにラッチの出力に接続されている請求項2に記載の並列疑似乱数列生成器。

(4) 前記直列疑似乱数列生成器を定義する直列排他的OR組合せが、6番目及び7番目の先行の直列出力を組合せるものであり、前記ラッチの数が16であり、前記ラッチは対応する出力Q0～Q15を有しており、該並列疑似乱数列生成器の幅が16に等しく、更に、各々が対応するラッチの入力に接続されている出力を有する対応する16個の排他的ORゲートEx0～Ex15の入力が、

Ex0の入力がQ8及びQ12に接続され、

Ex1の入力がQ9及びQ13に接続され、

Ex2の入力がQ10及びQ14に接続され、

Ex3の入力がQ11及びQ15に接続され、

Ex4の入力がQ0及びQ2に接続され、

Ex5の入力がQ1及びQ3に接続され、

Ex6の入力がQ2及びQ4に接続され、

Ex7の入力がQ3及びQ5に接続され、

Ex8の入力がQ4及びQ6に接続され、

Ex9の入力がQ5及びQ7に接続され、

Ex10の入力がQ0及びQ1に接続され、

Ex11の入力がQ1及びQ2に接続され、

Ex12の入力がQ2及びQ3に接続され、

Ex13の入力がQ3及びQ4に接続され、

Ex14の入力がQ4及びQ5に接続され、

Ex15の入力がQ5及びQ6に接続される

ようにラッチの出力に接続されている請求項1に記載の並列疑似乱数列生成器。

3. 発明の詳細な説明

<u>技術分野</u>

本発明は、直列疑似乱数列生成器(PRG)またはスクランブラ(scrambler)の出力を、直列PRGの連続する直列出力を与える複数の出力を備えた並列実現態様によりエミュレートするための回路及びその方法に関する。本発明は特に、適正な刻時(clocking)を保証するため及びデータストリームの潜在的な機密保護のために、高速データストリームを直列PRGと組み合わせる通信において使用される。このような通信データは高速であるが故に、直列PRGを相補形金属酸化物シリコン(CMOS)回路を使用して実現することはできない。従って、回路のクロック速度がCMOS回路の動作周波数内であるように、直列PRGをエミュレートする必要がある。

<u>発明の背景</u>

同期光網仕様(SONET)が採用されてから高速ディジタル通信に対する標準が設けられた(参照: American National Standards Institute,Inc. "Digital Hierarchy Optical Interface Rates and Formats Specification" 標準 T1.105-1988)。典型的にはこのようなディジタル通信は、データストリームが多数の隣合った0と1とで構成されている場合に、そうしなければ生じ得るであろうクロック信号の損失の可能性を最小化するために、疑似乱数列の直列スクランブル信号をデータストリームと組み合わせる。しかしながら直列データストリームは1秒当たり155メガビットまたはそれ以上で動作し得るので、直列PRGは、装置がより安価であり且つ対応するエミッタ結合論理(ECL)またはヒ化ガリウム(GaAs)回路より低い電力で動作

する好ましいCMOS回路ではなくて、個別のECL回路、ECL適用業務特定集積回路(ECL ASIC)またはヒ化ガリウム回路のような高速製造技術を使用して実現される必要がある。ECL及びGaAs回路の付加製造コスト及び電力要求は、付加的な熱を散逸するためにプリント回路基板面積をより大きくする必要があり、またもやCMOS回路、特にCMOS適用業務特定集積回路(CMOS ASIC)が好ましくなる。

CMOS回路は典型的には50メガビットより速いクロック速度で動作できないので、直列疑似乱数列生成器のクロック周波数を効果的に小さくするためにはある技術を使用することが必要である。本発明は、かかる技術と、任意の直列PRGの生成多項式に対して、及び等価の直列シフトレジスタの長さよりも大きい任意のサイズの、直列PRGからの連続出力を与える並列出力ワードに対して動作可能の回路を提供する。

こうして、比較的コストが低く電力消費量がより小さいCMOS回路を、直列PRGの出力をエミュレートする並列PRGを製造するために使用することができる。

<u>発明の概要</u>

並列疑似乱数列生成器は、直列PRGの次の入力値が直列PRGの複数の先行の出力の排他的OR(XOR)の組合せに等しい帰還構成において次々と動作する直列疑似乱数列生成器をエミュレートするものと説明される。例えば通信において典型的なスクランブル多項式は$1+X^4+X^7$である。この多項式は、直列PRGの次の入力値が、該生成器の5番目の先行の値と該生成器の7番目の先行の値とを排他的OR演算した出力に等しいことを意味する。生成器の7番目の値の出力も典型的には、スクランブルされるべきデータと排他的OR演算されている。

もし直列PRGがクロック速度fsを有するならば、並列PRGはクロック速度fp＝fs/H〔Hは並列PRGの出力の数である〕を有する。

並列PRGは、直列PRGを効果的にエミュレートする帰還経路を選択することにより任意の数の出力(任意の大きさのH)に拡張することができる。帰還経路は直列生成多項式と並列PRG実現態様の出力の大きさとに基づいている。それぞれHが8及び16の本発明の2つの好ましい実施例においては、対応する数のD型フリップフロップ(FF)を、シミュレートされる直列PRGの次のH個の連続する値に対応する次のH個の出力の値を決定するのに必要な帰還を提供する排他的OR(XOR)ゲートと一緒に使用する。これら2つの実現態様は、直列疑似乱数列生成器をシミュレートするため最小数の排他的ORゲートに対して設定された最適化基準を使用することにより最適化される。

<u>発明の目的</u>

本発明の目的は、H個の並列出力が直列疑似乱数列生成器のH個の連続出力値をエミュレートするための、直列疑似乱数列生成器の出力をシミュレートする並列疑似乱数スクランブラ回路とその方法とを提供する。

本発明の別の目的は、得られる並列クロック周波数が任意に低く設定され得、従ってCMOS製造技術を使用して並列PRGの実現態様を提供し得るようにHの値を任意の大きさにすることができる前記並列PRGを提供することである。

本発明の更に別の目的は、次のH個の出力を決定するためにH個の出力から必要な帰還を提供するための排他的ORゲートと一緒にD型フリップフロップを組み込んだ前記並列PRGを提供することである。

本発明の更に別の目的は、任意の直列PRG生成多項式に対して実行可能な前記並列PRGを提供することである。

本発明の他の目的は以下に明らかとなるであろう。

本発明の特徴及び目的がより充分に理解される

ように、添付の図面を参照して以下に本発明を詳細に説明する。

実施例

従来から通信情報をスクランブルするためには直列疑似乱数列生成器を使用する必要があった。第1図に示したように、典型的な直列疑似乱数列生成器20(直列PRG)にはシフトレジスタ22として構成された複数の段が、各段における値が次の段へと最後の段に遅延するまで伝送されるように組み込まれている。最後の段における値は典型的には、通信データストリームの1つのビットと排他的OR(XOR)演算され、XOR演算の結果が通信業務において実際に伝送される。排他的OR演算は、もし両入力が論理値1または論理値0であるならば出力は論理値0であり、もし入力がそれぞれ論理値1と論理値0またはこの逆であるならば出力は論理値1であると定義される。排他的OR演算を表わす真理表を表1に示す。

表1

2つの入力に対する排他的ORゲートの真理表

X_1	X_2	f
0	0	0
0	1	1
1	0	1
1	1	0

X_1及びX_2は入力であり、fは出力である。

(これは、繰上げのない2を法(モジュロ)とする和算に等価である。)

ほとんどの通信適用業務における疑似乱数列生成器の目的は、通信ビットストリームパターンとは無関係に、伝送される実際の情報がおおよそ同じ数の1と0とを含むことを保証することである。こうすると、例えばもし通信ビットストリームが長く連続する1または0のパターンを含むならばより困難になるであろう通信ビットストリームにおけるクロック情報の保守管理が容易となる。このようなスクランブルはデータ暗号化にも有効である。

再度第1図を参照すると、直列疑似乱数列生成器の動作は典型的には多項式:

$$1+X^n+\cdots+X^p$$

で定義され得ることが判る。これは、「+」が排他的OR演算を意味する特性多項式として公知である(本明細書中では全てこのように使用するものとする)。

この特性多項式に関係する帰還方程式は、

$$X^0+X^n+\cdots X^p=0$$

として誘導される。上記式から、

$$X^0+X^0+X^n+\cdots+X^p=0+X^0$$

となるが、一般に「+」が排他的OR演算子であると$X+X=0$及び$0+X=X$であるので、

$$X^n+\cdots+X^p=X^0$$

となる。この方程式は、シフトレジスタの次の入力値が$X^n+\cdots+X^p$であることを意味する。

例えば同期光網(SONET)標準(American National Standards Institute(ANSI)標準 T1.105-1988としても公知である)においてはこの多項式は$1+X^6+X^7$である。第1図から判るように、この多項式は、シフトレジスタ6における値がレジスタ7における値と排他的OR演算され、その結果がP段シフトレジスタの段1における次の値となることを意味する。表4は、7段シフトレジスタの各段に対する出発値が論理値1である場合の7つの段における値を示す。この出発値は典型的には「シード(seed)」と称される。SONET標準においては、シードは典型的には直列PRGに対して全て1である。これから判るように、段1に対して生成される値はシフトレジスタの段を連続的に移動する。前記したように、段7からの出力は直列通信ビットストリームと排他的OR演算するためにも使用される。

このような生成器が疑似乱数列生成器と称される理由は、生成されるビットストリームが同じ出

発シード及び同じ多項式に対して常に同じであるからである。

SONET多項式は段6及び7の排他的OR演算を使用したが、勿論、直列シフトレジスタの異なる段を合わせて排他的OR演算する他の多項式を使用することもできる。実際、所望であれば2つ以上の段を排他的OR演算してもよい。

一般には最大長の多項式、即ち最大カウント(クロックサイクル)後にそれ自体を繰り返す多項式が使用される。最大長の多項式に対するカウントの最大数は、n次多項式において2^n-1である。例えば、3次の多項式では最大多項式は$1+X^2+X^3$であり、非最大多項式は$1+X^1+X^2+X^3$である。表2及び3から判るように、最大長多項式は7つの出力の後に繰り返し、一方、非最大長多項式は4つの出力の後に繰り返す。

本発明は、最大であろうとなかろうと任意の直列多項式を用いて適用可能である。

表2

次数3の多項式

X^2+X^3

(最大長$=2^n-1=2^3-1=8-1=7$)

	直列段番号		
(クロックサイクル)	1	2	3
1	0	1	1
2	0	0	1
3	1	0	0
4	0	1	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	1	1
9	等		

表3

次数3の多項式

$X^1+X^2+X^3$

	直列段番号		
(クロックサイクル)	1	2	3
1	0	1	1
2	0	0	1
3	1	0	0
4	1	1	0
5	0	1	1
6	等		

このような直列疑似乱数列生成機は、通信ビットストリームの伝送速度が約50メガビット/秒を越えると集積回路の実現態様に問題が生じる。50メガビット/秒を越える速度では、相補形金属酸化物シリコン(CMOS)集積回路の製造は非実用的になる。実際に、使用可能速度が約75メガヘルツを越えるCMOSの製造は実質的に不可能である。結果としてSONET標準に使用されるもののような高い伝送速度(例えば155メガビット/秒)に対しては、このような直列疑似乱数列生成機を使用するのであれば、エミッタ結合論理(ECL)またはヒ化ガリウム(GaAs)技術を使用して製造することが必要となる。上記両技術はCMOS技術と比較して、典型的には製造がより難しく、より多くの熱を生成するので発生した熱を散逸するための集積回路素子を設置するためにより大きな面積のプリント回路基板を必要とし、1論理ゲート当たりのコストがより高くなるという深刻な欠点を有する。

本発明は、その値が直列疑似乱数列生成機の連続出力を与える複数の並列出力を有する並列疑似乱数列生成機を提供することにより、高速度疑似ランダムビットパターンの生成に対する一般的な解決策を提供する。かかる並列疑似乱数列生成機24は任意の所望の数の並列出力を有することができ、第2図に示した実施例では8個の出力を有しており、第4図に示した実施例では16個の出力を有している。並列疑似乱数列生成機のサイズは、並列ワードのサイズがスクランブル多項式の次数に等しいかまたはそれより大きい限りは、特定の適用業務に最も適した任意の値に設定することができる。ディジタル集積回路を使用する場合には、出力の数は一般に2の倍数に等しい値を有し、例えば8個、16個等の出力を有する。

第2図に示した実施例においては、疑似乱数列生成機は、その出力(Q0～Q7)が、エミュレートされる直列疑似乱数列生成機の8個の連続出力値を与える8個のラッチ26を有する。ラッチ26はD型のフリップフロップとすることができる。直列疑似乱数列生成機の出力が直列段7である表4を参

照すると、この7番目の段は最初の7つの直列クロックサイクル(直列クロックサイクル0～6)に対しては論理値1を有し、次のクロックサイクル(直列クロックサイクル7)に対しては論理値0を有することが判る。従って8ビット並列PRGの出力Q7～Q0は、表5に示したように直列PRGにおける段7の8個の連続出力値を与えることができる。即ち表5から、Q0出力はこの直列PRG出力段7の8番目の直列出力を与え、Q1は段7の7番目の直列出力を与える等、同様にQ7までこの直列PRGの段7の最初の直列出力を与える。このパターンが新たな並列出力の各々に対して繰り返される。

以下に記載するように、並列PRGの次の8つの出力Q7～Q0は値00000100を有する。これらQ7～Q0に対する値は、表4に表された段7の時間出力8～15を、並列クロックサイクル1に対する並列出力(表5参照)と比較することから判るように、直列段7の次の8個の直列出力を与える。従って並列PRGの最初(0番目)のフレームは直列PRGの段7の最初の8つの直列出力(直列クロックサイクル)に対応しており、並列PRGのフレーム1は段7の次の8つの直列出力(直列クロックサイクル8～15)に対応する等となる。最初の並列フレームは、それ自体が特定の出発シーケンスまたはシードを有する直列PRGをエミュレートする上でその動作を開始するための、生成機への並列シード入力である。

表4

1+X⁶+X⁷生成多項式に対応する直列疑似乱数列発生機

時間(等価の直列クロックサイクル)	直列段番号 1	2	3	4	5	6	7	
0	1	1	1	1	1	1	1	
1	0	1	1	1	1	1	1	
2	0	0	1	1	1	1	1	
3	0	0	0	1	1	1	1	並列フレームA0 (8ビットの場合)
4	0	0	0	0	1	1	1	
5	0	0	0	0	0	1	1	
6	0	0	0	0	0	0	1	
7	1	0	0	0	0	0	0	
8	0	1	0	0	0	0	0	並列フレームA0 (16ビットの場合)
9	0	0	1	0	0	0	0	
10	0	0	0	1	0	0	0	
11	0	0	0	0	1	0	0	並列フレームA1
12	0	0	0	0	0	1	0	
13	1	0	0	0	0	0	1	
14	1	1	0	0	0	0	0	
15	0	1	1	0	0	0	0	
16	0	0	1	1	0	0	0	
17	0	0	0	1	1	0	0	
18	0	0	0	0	1	1	0	
19	1	0	0	0	0	1	1	並列フレームA2
20	0	1	0	0	0	0	1	
21	1	0	1	0	0	0	0	
22	0	1	0	1	0	0	0	
23	0	0	1	0	1	0	0	並列フレームA1
24	0	0	0	1	0	1	0	
25	1	0	0	0	1	0	1	
26	1	1	0	0	0	1	0	
27	1	1	1	0	0	0	1	並列フレームA3
28	1	1	1	1	0	0	0	
29	0	1	1	1	1	0	0	
30	0	0	1	1	1	1	0	
31	1	0	0	1	1	1	1	

表4(続き)

1+X⁶+X⁷生成多項式に対応する直列疑似乱数列発生機

時間(等価の直列クロックサイクル)	直列段番号 1	2	3	4	5	6	7	
32	0	1	0	0	1	1	1	
33	0	0	1	0	0	1	1	
34	0	0	0	1	0	0	1	
35	1	0	0	0	1	0	0	並列フレームA4
36	0	1	0	0	0	1	0	
37	1	0	1	0	0	0	1	
38	1	1	0	1	0	0	0	
39	0	1	1	0	1	0	0	
40	0	0	1	1	0	1	0	
41	1	0	0	1	1	0	1	
42	1	1	0	0	1	1	0	
43	1	1	1	0	0	1	1	並列フレームA5
44	0	1	1	1	0	0	1	
45	1	0	1	1	1	0	0	
46	0	1	0	1	1	1	0	
47	1	0	1	0	1	1	1	
48	0	1	0	1	0	1	1	
49	0	0	1	0	1	0	1	
50	1	0	0	1	0	1	0	
51	1	1	0	0	1	0	1	並列フレームA6
52	1	1	1	0	0	1	0	
53	1	1	1	1	0	0	1	
54	1	1	1	1	1	0	0	
55	0	1	1	1	1	1	0	

並列フレームA2 (16ビットの場合)

表5

1+X⁶+X⁷生成多項式に対応する直列PRGをエミュレートする並列疑似乱数列生成機(幅=8ビット)

並列クロックサイクル	直列クロックサイクル	最終出力 Q0	Q1	Q2	Q3	Q4	Q5	Q6	最終出力 Q7
0	0－7	0	1	1	1	1	1	1	1
1	8－15	0	0	1	0	0	0	0	0
2	16－23	0	0	0	1	1	0	0	0
3	24－31	1	0	0	0	1	0	1	0
4	32－39	0	0	1	0	0	1	1	1
5	40－47	1	0	0	1	1	0	1	0
6	48－55	0	0	1	0	1	0	1	1

直列PRGをシミュレートするための並列実現態様解析

第2図から判るように、ラッチ26に加えて並列PRGには更に複数の排他的ORゲート28が組み込まれており、排他的ORゲート28は、ラッチが次の出力を生成するようにラッチへ入力として与えるためにラッチの種々の出力を組合わせる。第3図は生成機を起動し(ANDゲート34)、並列シードをロードし(ORゲート36)、且つ並列クロック信号38を与えるための追加論理回路を示す第2図に対応する概略図である。

第2図から判るように、8つのフリップフロップの入力D0～D7には関数F0～F7に関係する値が与えられる。これらの関数は表5Aに示した方程式で定義される。

表5A

$F0 = Q4 + Q6$

$F1 = Q5 + Q7$

$F2 = Q0 + Q1$

$F3 = Q1 + Q2$

$F4 = Q2 + Q3$

$F5 = Q3 + Q4$

$F6 = Q4 + Q5$

$F7 = Q5 + Q6$

第2図においては更に並列疑似乱数列生成機の出力Q0～Q7は、対応する数のデータストリーム出力排他的ORゲート30に次々と与えられることが判る。排他的ORゲート30においては、各排他的ORゲートへの第2の入力には、Q7の排他的OR出力ゲート30'への入力は直列データの最初のビットと排他的OR演算され、出力Q6は直列データの次のビットと排他的OR演算される等、Q0が直列データの8番目のビットと排他的OR演算されるように、直列データストリームの1つのビットが与えられる。従って出力線32における出力信号は、8ビットマルチプレクサ(図示なし)を使用することにより直列ビットストリームに変換し直され得るスクランブルされた出力データを与える。

第2図から、もし並列疑似乱数列生成機が幅8(N=8)を有するならば、シミュレートされた直列疑似乱数列生成機の次の8つの出力をそれぞれQ7～Q0で与えられるように各々の並列計算がなされるので、並列動作の周波数は到来する直列ビットストリームのものの8分の1となることは容易に判る。

並列出力排他的OR組合せの決定

以下により充分に説明するように、並列疑似乱数列生成機の各ラッチへの入力を与えるための排他的ORゲート構成は、直列疑似乱数列生成機の出力ビットストリームをエミュレートするように決定される。第2図には特定の排他的ORゲート構成を示してあるが、実際には多くの実現態様が可能である。本発明は、各入力に対して最小数の排他的ORゲートを使用する場合に特に有利である。こ

の構成は、直列ゲートに対する必要条件を最小化し、結果として各直列ゲートに付随するゲート遅延を最小化する。

任意の疑似乱数列生成機の多項式に対して、並列PRFの幅が、直列PRGの多項式を定義するのに使用される最大シフトレジスタ段に少なくとも等しいという条件で、排他的ORゲートを並列疑似乱数列生成機を実行するために使用することができる解決策が存在することは、経験的に見出されており、本明細においても発明者C.Rogerによる標題"並列疑似乱数列発生機、数学的解析(Parallel Psedo-Random Generator,Mathematical Analysis)"の数学的解析のなかに記載されているように数学的検証もされている。

第1図に関して与えられた上記多項式、即ち段1への次の入力が段6及び7の排他的OR出力に等しい多項式においては、この関係は一般に、

$$Q(n) \equiv Q(n+6) + Q(n+7) \qquad (1)$$

〔式中nは直列PRGの任意の段である〕

と定義され得ることが判る。第7図はこの関係を図で表したものである。

再度表4を参照すると、クロックサイクル0に対する段6及び7がともに論理値1を有することが判る。その結果、段1に対する次の値は0となる(1+1=0(表1参照))。この結果は、nが0である場合の上記式と同等となる(次の直列クロックサイクル後にはQ(0)はQ(1)となり、一般に次の直列クロックサイクル後にはQ(n-1)はQ(n)となる)。

エミュレーション用8出力並列疑似乱数列生成機の次の8ビットを決定するためには、直列PRGの次に生成されるビットが、8直列クロックサイクロ後に並列PRGの出力Q7に対する次の値となることが判る(Q-1は、1直列クロックサイクル後にQ0となり、更に7直列クロックサイクル後にQ7となり、これら8つの直列クロックサイクルは1つの並列ワードクロックサイクルと等価である(表6参照))。即ち8ビット並列PRGの実現態様においては、Q7に対する次の値はQ-1に等しく、従ってQ5及びQ6の排他的ORに等しいことになる。即ち、

次のQ7=F7=Q5+Q6

である。

この同じ関係を用いてQ6～Q2の次の値は以下のように定義され得る。即ち、

次のQ6=F6=Q4+Q5、

次のQ5=F5=Q3+Q4、

次のQ4=F4=Q2+Q3、

次のQ3=F3=Q1+Q2、

次のQ2=F2=Q0+Q1

となる。

Q1の数値の求め方は表6及び第7図を参照することにより最も良く理解され得る。

表6

2つの8ビットワードに対する並列出力値
(n=-8～n=7)

Q-8	Q-7	Q-6	Q-5	Q-4	Q-3	Q-2	Q-1	Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7
次の8ビットワード								現在の8ビットワード							

従って、(F1と等価の)Q1に対する次の値はQ-7の値に等しい。即ち、

次のQ1=F1=Q-7

である。式(1)から、

$$F1 = Q(-7+6) + Q(-7+7) = Q-1 + Q0 \quad (n=-7)$$

となるが、(式(1)を再度使用すると)、

$$Q-1 = Q5 + Q6 \quad (n=-1)$$

であるので、

次のQ1=F1=Q-1+Q0=Q5+Q6+Q0

と書ける。しかしながら更に式(1)から、Q0の現在の値はQ6+Q7の現在の値に等しい(Q0=Q6+Q7)ので、

次のQ1=F1=Q5+Q6+Q6+Q7 (2)

となる。全ての論理値のそれ自体の排他的ORは0となるので(表1参照)、式(2)は、

次のQ1＝F1＝Q5＋Q7

と書き換えることができる。

同じ原理を使用し、Q0の次の値は、

次のQ0＝F0＝Q4＋Q6

と定義されることは容易に明らかである。

従って、排他的ORゲート構成を決定するための方法に不可欠なことは、直列生成多項式によって直列段間の相関関係を決定することである。並列関係は単に複数の直列段を同時に与えるものであるので、Nを並列出力の幅(即ち数)とするとN個の直列クロックサイクル後の並列出力の各々に対して次の並列出力を計算するために、直列多項式を使用する。並列出力段の現在の値のみが、かかる同じ段の次の値を計算するために使用可能であるので、出力値が並列PRGの1つ以上の次の出力(表6に示した次のワード)から要求されるのであれば、次の出力値を与える現在の出力を決定する特定の出力に対して直列多項式を再度使用する。この方法はいかなるサイズの並列ワードに対してもまたいかなる直列生成多項式に対しても使用することができる。

表7を参照すると、16ビット並列PRGの実現態様において、Q15の次の値は単にQ-1(即ち16直列クロックパルス後の直列出力)に等しい。即ち、

次のQ15＝F15＝Q-1＝Q5＋Q6　(n＝-1に対する式(1)参照)

である。

表7

16ビット並列PRG

	-16	-15	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1
次の16ビットワード	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q
現在の16ビットワード	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

この解析はQ14～Q10の次の値に対しても成立する。即ち、

次のQ14＝F14＝Q4＋Q5

次のQ13＝F13＝Q3＋Q4

次のQ12＝F12＝Q2＋Q3

次のQ11＝F11＝Q1＋Q2

次のQ10＝F10＝Q0＋Q1

である。Q9の次の値はF9、つまりQ-1＋Q0に等しいことが判るが、Q-1は単にQ5＋Q6に等しいので、

次のQ9＝F9＝Q5＋Q6＋Q0

となる。Q0の現在の値は式(1)によってQ6＋Q7に等しく、従って、

次のQ9＝F9＝Q5＋Q6＋Q6＋Q7

次のQ9＝F9＝Q5＋Q7

となる。同様にQ8～Q4の次の値に対しては、

次のQ8＝F8＝Q4＋Q6

次のQ7＝F7＝Q3＋Q5

次のQ6＝F6＝Q2＋Q4

次のQ5＝F5＝Q1＋Q3

次のQ4＝F4＝Q0＋Q2

である。

Q3の次の値はQ-13に等しい。上記式(1)を使用すると、

次のQ3＝F3＝Q-13

Q-13＝Q-7＋Q-6　　(n＝-13)

Q-13＝(Q-1＋Q0)＋(Q0＋Q1)

Q-13＝Q-1＋Q1

Q-13＝(Q5＋Q6)＋Q1

Q-13＝(Q5＋Q6)＋(Q7＋Q8)　　(2a)

であり、更に式(1)から、

Q5＝Q11＋Q12

Q6＝Q12＋Q13

Q7＝Q13＋Q14

Q8＝Q14＋Q15

であり、従って、

Q-13＝(Q11＋Q12)＋(Q12+Q13)＋(Q13＋Q14)＋(Q14+Q15)

となる。即ち、

Q3＝Q11＋Q15　　(2b)

である。同様に、

次のQ2＝F2＝Q10＋Q14

次のQ1＝F1＝Q9＋Q13

次のQ0＝F0＝Q8＋Q12

である。

上記解析に対応する16ビット並列PRGに対する排他的OR演算実現態様は第4図に示してある。

次のQ13は式(2a)及び(2b)によって示されるような複数の排他的OR演算によって定義され得ることが判る。一般に出力に対してはこのような多重表示が与えられ得る。1つの最適化基準は、出力Q3に対して式(2b)で示されるような最小数のゲート入力を使用することであろう。

上記解析は、並列PRGの幅がエミュレートされる直列PRGに対する帰還構成において使用される最大数の直列段に少なくとも等しいという条件で任意の幅の並列PRGに対して使用することができる。直列多項式が、次の入力段を計算するために段6及び7を使用する上記実施例においては、Pの値は7であり、即ち並列PRGの幅は少なくとも7に等しい必要があるが、これより大きくともよい。

更に、直列多項式は2つの直列段の排他的ORに等しいとしたが、本発明は、次の入力を計算するために排他的OR演算される直列段の数とは無関係に、任意の直列多項式に適用可能である。

より一般的な直列疑似乱数列生成機の例では多項式：

次の直列入力＝$X^2+X^3+X^7$

を使用する。即ち特性多項式は$1+X^2+X^3+X^7$である。

この多項式は非最大(前記表2及び3参照)であるが、並列PRG実行方法が適用業務において一般的であること示すために与えられている。

第8図は、段nに関して、

Q(n)≡Q(n＋2)＋Q(n＋5)＋Q(n＋9)　　(3)

であるこの並列疑似乱数列生成機を示す。

表8は、この多項式の36のクロックサイクル(クロックサイクル0～35)に対応する直列疑似乱数列生成機を構成する9つの段に対する直列段の値を示す。

表8

直列多項式 $= 1 + X^2 + X^5 + X^9$

(クロックサイクル)	直列段番号 1	2	3	4	5	6	7	8	9	
0	0	1	1	1	1	1	1	1	1	
1	1	0	1	1	1	1	1	1	1	
2	0	1	0	1	1	1	1	1	1	
3	1	0	1	0	1	1	1	1	1	並列フレームQ
4	0	1	0	1	0	1	1	1	1	(9ビットの場合)
5	0	0	1	0	1	0	1	1	1	
6	0	0	0	1	0	1	0	1	1	
7	1	0	0	0	1	0	1	0	1	
8	0	1	0	0	0	1	0	1	0	
9	1	0	1	0	0	0	1	0	1	
10	1	1	0	1	0	0	0	1	0	
11	1	1	1	0	1	0	0	0	1	
12	1	1	1	1	0	1	0	0	0	
13	1	1	1	1	1	0	1	0	0	並列フレーム1
14	0	1	1	1	1	1	0	1	0	
15	0	0	1	1	1	1	1	0	1	
16	0	0	0	1	1	1	1	1	0	
17	1	0	0	0	1	1	1	1	1	
18	0	1	0	0	0	1	1	1	1	
19	0	0	1	0	0	0	1	1	1	
20	1	0	0	1	0	0	0	1	1	
21	1	1	0	0	1	0	0	0	1	
22	1	1	1	0	0	1	0	0	0	並列フレーム2
23	1	1	1	1	0	0	1	0	0	
24	1	1	1	1	1	0	0	1	0	
25	0	1	1	1	1	1	0	0	1	
26	1	0	1	1	1	1	1	0	0	
27	1	1	0	1	1	1	1	1	0	
28	0	1	1	0	1	1	1	1	1	
29	1	0	1	1	0	1	1	1	1	
30	1	1	0	1	1	0	1	1	1	
31	1	1	1	0	1	1	0	1	1	
32	1	1	1	1	0	1	1	0	1	並列フレーム3
33	0	1	1	1	1	0	1	1	0	
34	0	0	1	1	1	1	0	1	1	
35	0	0	0	1	1	1	1	0	1	

式(3)の関係を使用することにより、第8図に示した直列疑似乱数列生成機をエミュレートする幅M＝9を有する並列疑似乱数列生成機の次の出力Q0～Q8に対する値は以下のようになる。即ち、

次のQ8 = F8 = Q-1 = Q1 + Q4 + Q8

次のQ7 = F7 = Q-2 = Q0 + Q3 + Q7

次のQ6 = F6 = Q-3 = Q-1 + Q2 + Q6

= Q1 + Q4 + Q8 + Q2 + Q6

= Q1 + Q2 + Q4 + Q6 + Q8

次のQ5 = F5 = Q-4 = Q-2 + Q1 + Q5

= Q0 + Q3 + Q7 + Q1 + Q5

= Q0 + Q1 + Q3 + Q5 + Q7

次のQ4 = F4 = Q-5 = Q-3 + Q0 + Q4

= Q-1 + Q2 + Q6 + Q0 + Q4

= Q1 + Q4 + Q8 + Q2 + Q6 + Q0 + Q4

= Q1 + Q8 + Q2 + Q6 + Q0

= Q0 + Q1 + Q2 + Q6 + Q8

次のQ3 = F3 = Q-6 = Q-4 + Q-1 + Q3

= (Q-2 + Q1 + Q5) + (Q1 + Q4 + Q8) + Q3

= (Q0 + Q3 + Q7) + Q1 + Q5 + Q1 + Q4 + Q8 + Q3

= Q0 + Q7 + Q5 + Q4 + Q8

= Q0 + Q4 + Q5 + Q7 + Q8

次のQ2 = F2 = Q-7 = Q-5 + Q-2 + Q2

= (Q-3 + Q0 + Q4) + (Q0 + Q3 + Q7) + Q2

= ((Q-1 + Q2 + Q6) + Q0 + Q4) + (Q0 + Q3 + Q7) + Q2

= ((Q1 + Q4 + Q8) + Q2 + Q6) + Q0 + Q4)

+ (Q0 + Q3 + Q7) + Q2

= Q1 + Q8 + Q6 + Q3 + Q7

= Q1 + Q3 + Q6 + Q7 + Q8

次のQ1 = F1 = Q-8 = Q-6 + Q-3 + Q1

= (Q0 + Q7 + Q5 + Q4 + Q8)

+ (Q1 + Q4 + Q8 + Q2 + Q6) + Q1

= Q0 + Q7 + Q5 + Q2 + Q6

= Q0 + Q2 + Q5 + Q6 + Q7

次のQ0 = F0 = Q-9 = Q-7 + Q-4 + Q0

= (Q1 + Q8 + Q6 + Q3 + Q7)

+ (Q0 + Q3 + Q7 + Q1 + Q5) + Q0

= Q8 + Q6 + Q5

= Q5 + Q6 + Q8

表9は、直列クロックサイクル0～35に対応する4つの並列クロックサイクルに対する並列疑似乱数列生成機の出力値を示す。これらの出力は、最初の36の直列クロックサイクルに対する直列疑似乱数列生成機の出力段9に対応することが判る。

表9

1+X³+X⁵+X⁹多項式に対応する直列PRGをエミュレートする並列疑似乱数列生成機(幅=9ビット)

並列クロックサイクル	直列クロックサイクル	直列Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
0	0－8	0	1	1	1	1	1	1	1	1
1	9－17	1	0	1	0	0	0	1	0	1
2	18－26	0	1	0	0	0	1	1	1	1
3	27－35	1	1	0	1	1	1	1	1	0

以上の説明から、疑似乱数列生成機の幅が直列疑似乱数列生成機に使用されている段の数に少なくとも等しい限りは、並列疑似乱数列生成機が実現可能であることが判る。更に、並列PGRを実行するために必要な排他的ORゲートの最小数は、少なくとも並列PGRが直列PRGに等しい幅を有する場合には、対応する直列PRGに使用されている排他的ORゲートの数に必ずしも等しくないことが判る。

以下数学的解析によって、並列PRGの幅が少なくとも直列PRG多項式を実現するのに必要な段の数に等しい場合に、直列PRGの並列PRGエミュレーションが常に存在することを立証する。

並列疑似乱数列生成機

数学的解析

序

並列疑似乱数列生成機について、シフトレジスタを用いて構築されている従来の直列PRG生成機を置き換えるための解析を行なう。並列及び直列の生成機をそれぞれ第5A図及び第5B図に示した概略図によって説明する。

従来の解決策においては、P段シフトレジスタの選つかの段から発信された信号は一緒に排他的OR(XOR)ゲートによって加えられ、得られた信号はレジスタの入力に供給され、帰還を生成する。連続する信号値の間には方程式:

$$Sn = A1\ Sn-1 + A2\ Sn-2 + \cdots\cdots + Ap\ Sn-p \qquad (4)$$

〔式中、「+」はXORまたは2を法とする加算に対して使用されており、A1,・・・・,Ai,・・・Apは、段iが接続されているならば1であり、そうでなければ0である〕

が成立している。これは、係数が2を法とする整数の体(フィールド)、即ち'F(0,1)'に含まれる方程式である。

信号を「Z変換」すると、

$$S(Z) = A1\ Z^1S(Z) + A2\ Z^2S(Z) + \cdots + Ap\ Z^pS(Z) \qquad (5)$$

または

$$P(Z)\cdot S(Z) = 0 \qquad (6)$$

但し: $P(Z) = Z^p + A_1Z^1 + A_2Z^2 + \cdots + ApZ^p$ (7)

となる。方程式(5)と(6)とは等価である。

P(Z)はSの特性多項式であり、Sの「生成式(generator)」と考えてもよい。

多項式P(Z)は「既約及び素」であり(係数がF(0,1)に含まれる、より小さい次数の多項式の積ではない)、原始根$Z^q + 1 = 0 (q = 2^p - 1)$を有し、系によって生成される数列は周期$2^p - 1$の疑似乱数列生成式となる。

並列生成機は、その入力信号がXORゲート網によって計算される多重出力ラッチ(例えば複数の

フリップフロップ)で構成されており、またラッチの出力信号はこのXORゲート網に供給される。

基本事項

以下の事項はディジタル信号処理方法を専門としない読み手に有効であろう。

1) 「2を法とする整数」の体はただ2つの元、即ち0と1とを含み、2種類の演算、即ち乗算(AND)と加算(排他的ORまたはXOR)とにおいて、

$0\times0=0$、$0\times1=1\times0$、$1\times1=1$、及び

$0+0=0$、$0+1=1+0=1$、$1+1=0$

が成立する。係数がこの体にある多項式は以下の特性を有する。即ち、

$P(Z)=Q(Z)$は$P(Z)+Q(Z)=0$と等価である、または$(1+Z)^2=1+Z^2$(何故ならば$2=0$である)である。

2) もし$S_n=S(nT)$(Tは時間間隔である)ならば、

$S(Z)=\Sigma S_nZ^n$ 〔Zは遅延演算子である〕

であるような「Z変換」を使用する。何故ならば、

$Z\,S(Z)+\Sigma S_nZ^{n+1}=\Sigma S_{n-1}Z^n$

であり、これはS(t-T)のZ変換であるからである。

3) $S_n=A_1S_{n-1}+A_2S_{n-2}+\cdots+A_pS_{n-p}$ (5)

の解を求める場合に、通常、

$Sn=C\,a^{-n}$ 〔Cは定数である〕

と置くと方程式(5)は、

$a^{-n}=A_1a^{-n+1}+A_2a^{-n+2}+\cdots+A_pa^{-n+p}$

または$a^p+A_1a+A_2a^2+\cdots+A_pa^p=0$

となり、「a」は$P(Z)=0$の根であるべきであることが判る。この方程式にはp個の根が存在する。

これらの根は一般に0または1で表わすことはできないが、式(5)の一般解はそれらの連続累乗の線形組合せとなる。即ち、

$S_n=a_1^{-n}+a_2^{-n}+\cdots+a_p^{-n}$

となり、これはP(Z)の根の対称関数、即ちもしこれらの係数が2を法とする整数の体に含まれるならば、0または1であるP(Z)の係数の関数である。

4) pビットを含む直列PNS生成機のシフトレジスタは、ヌル配列を生成するワード0,0,0…を含み多くとも2^pワードを包含することは容易に判る。従って多くとも2^p-1個のヌルでないワードが存在し得、配列の周期は多くとも2^p-1である。この周期は、「既約及び素」の多項式と称される特定の多項式を用いて得られる。

5) 「a」を$P(Z)=0$の根とすると、

$P(a)=a^0+A_1a^1+A_2a^2+\cdots+A_pa^p=$

$P^2(a)=a^0+A_1a^2+A_2a^4+\cdots+A_pa^{2p}$

である。これは、Ai=Aiであり、

且つ上記式のp個の根は$a\cdots a^{2^{p-1}}$であるので

$a^{2^{p-1}}$は$P(Z)=0$のもう1つの根であるが故である。

また$a^{2^{p-1}}=1$であるので累乗指数2^pを有する次の根は「a」に等しい。

並列生成機

第5A図の並列生成機の「Z方程式」は、

$Z^0=A_{0,0}Z^n+A_{0,1}Z^{n+1}+\cdots+A_{0,n-1}Z^{2n-1}$

$Z^1=A_{1,0}Z^n+A_{1,1}Z^{n+1}+\cdots+A_{1,n-1}Z^{2n-1}$

……

$Z^i=A_{i,0}Z^n+A_{i,1}Z^{n+1}+\cdots+A_{i,n-1}Z^{2n-1}$

……

$Z^{n-1}=A_{n-1,0}Z^n+A_{n-1,1}Z^{n+1}+\cdots+A_{n-1,n-1}Z^{2n-1}$

である。

(その要素が$A_{i,j}$である)行列はラッチの2つの連続する状態n-1とnとの間の遷移行列(transition matrix)である。これらの係数は、出力jが入力iに一般にXOR回路によって連結されているか否かに応じて1または0となる。例えば方程式iは、

$S_{n-i}=(A_{i,0}S_{n-n})XOR(A_{i,1}S_{n-n-1})XOR(A_{i,2}S_{n-n-2})\cdots\cdots XOR(A_{i,n-1}S_{n-2n+1})$

に対応しており、方程式iは、

$Zi=\sum_{j=0}^{N-1}A_{i,j}Z^{n+j}$ または

$Z^0=Z^{n-1}\sum_{j=0}^{N-1}A_{i,j}Z^j=Z^{n-1}R_i(Z)$

と書くことができる。

R_iは、その係数が遷移行列の行iの要素である多項式である。

$T_i(Z)=Z^n+Z^{n-i}R_i(Z)$は、$T_i(Z)\cdot S(Z)=0$であるものとなるべきである。$P(Z)\cdot S(Z)=0$は既知であるので、$T(Z)$がPの倍数であるならば、例えば$T(Z)=P(Z)\cdot Q(Z)$であり、すると、

$$T(Z)\cdot S(Z)=P(Z)\cdot Q(Z)\cdot S(Z)$$
$$=Q(Z)[P(Z)\cdot S(Z)]=0$$

となる。

(この結果は、Sの連続する値が$P(Z)=0$の根の累乗の組合せであることを考慮することにより得ることができ、これは、$T(Z)=0$が少なくともP(Z)=0と同じ根を有することを示唆する)。

ここで、多項式:

$$P=A_0+A_1Z+A_2Z^2+A_3Z^3$$

を考える(理解し易いように特定の例を用いるが、誘導方法は一般的である)。P(Z)によって生成される数列は、

$$A_0S_{n-3}+A_1S_{n-2}+A_2S_{n-1}+A_3S_n=0$$
$$A_0S_{n-4}+A_1S_{n-3}+A_2S_{n-2}+A_3S_{n-1}=0$$
$$A_0S_{n-5}+A_1S_{n-4}+A_2S_{n-3}+A_3S_{n-2}=0$$
$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

である。$Q(Z)=B_0+B_1Z+B_2Z^2$と置くと、

$$T(Z)=P(Z)\cdot Q(Z)=A_0B_0+(A_0B_1+A_1B_0)Z+$$
$$(A_0B_2+A_1B_1+A_2B_0)Z^2+(A_1B_2+A_2B_1+A_3B_0)Z^3$$
$$+(A_2B_2+A_3B_1)Z^4+A_3B_2Z^5$$、または

$$T(Z)=C_0+C_1Z+C_2Z^2+C_3Z^3+C_4Z^4+C_5Z^5$$

となる。

計算するとS'は数列であり、

$$C_0S'_{n-5}+C_1S'_{n-4}+\cdots+C_5S'_n$$
$$=A_0B_0S'_{n-5}+(A_0B_1+A_1B_0)S'_{n-4}+\cdots$$

となる。従って、

$$B_0(A_0S'_{n-5}+A_1S'_{n-4}+A_2S'_{n-3}+A_3S'_{n-2})+$$
$$B_1(A_0S'_{n-4}+A_1S'_{n-3}+A_2S'_{n-2}+A_3S'_{n-1})+$$
$$B_2(A_0S'_{n-3}+A_1S'_{n-2}+A_2S'_{n-1}+A_3S'_n)$$

であり、この式は、S'がP(Z)によって生成される数列であればOとなる。更に、もしS'_n、S'_{n-1}、S'_{n-2}、S'_{n-3}がSの連続する値であるならば、計算した和はOとなり、S'_{n-5}はSの次のサンプルとなる。以上から、もしT(Z)がP(Z)の倍数であるならば、

1)期待通り$T(X)\cdot S(Z)=0$であり、且つ

2)「良いシード」が与えられたならばT(Z)は数列Sを生成すると結論される。この結果は、数列Sの一連のサンプルを意味する(別のシードを用いてもQによって数列が生成されるであろう)。

T(Z)では、(常に1である)その最初の係数とそのN番目の最後の係数(R_iの係数)のみがゼロでないとすることができる。従って、必要とされるシードはラッチ内に含まれるサンプルに制限される。出発時には、ラッチにP(Z)によって生成される数列の一部を負荷することが必要である。全ての多項式$T_i(Z)$はラッチの次のシリーズのビットのサンプルを生成し、各クロック時に対して図の左側に位置するNビットシリーズ(状態n)は右側に位置するシリーズ(状態n-1)となる等が言える。

第6図は多項式$T_i(Z)$の係数の図である。1に等しい係数は「X」で表してあり、それ以外は0に等しい。遷移行列の要素であるR_iの係数は平行四辺形の内側にあり、問題となるのは、

1)P(Z)の倍数、またはP(Z)によって生成される数列の生成式(これは等価である)であり、

2)最初の係数を除いて平行四辺形に包含される係数を有し、

3)最も単純な実行態様を与えるために最小の1に最小の項を有する

多項式を見付けることであることが判る。

「Bezoutの関係」(表10参照)は上記最初の2つの条件を満足する多項式を計算することができるが、第3の条件は常に満足するとは限らない。

「良い」多項式を見付ける極めて単純な方法は、行列の各行、即ち平行四辺形内に含まれる2つの

ゼロでない係数を有する多項式に対して「試算及び検証(try and see)」を行なうことである。これらの多項式は数列Sの生成式としてテストされ、最初に見付けられたものを使用し、可能であれば行列の次の行に対してもそれを再度使用する。2つの係数を用いた検索が失敗した場合には3つまたはそれ以上の係数に対して検索する等となる。

多項式は本来単純であるけれども、数10秒間の計算時間を要することがある(12次の多項式、N=32に対して20秒(D.E.C. VAX8600コンピュータ)を要するが、7次の多項式でN=8または16に対してはほぼ即座に結果がでる)。ここでNはW、即ち並列出力の数と等価である。

<u>他の誘導方法</u>

a)並列システムの特性方程式

かかる並列生成機は、その要素が0または1(1はXOR演算を意味する)である遷移行列によって連結されているラッチの2つの連続する状態によって与えられ得る。そして、第2の状態の信号の各々は第1の状態の信号に式:

$$Z^{i}=\Sigma_j B_{ij} Z^{j\cdots} \quad (8)$$

〔$0\leqq i\leqq N-1$且つ$0\leqq j\leqq N-1$, $B_{ij}=0$または1〕によって依存する。Nはラッチ内に含まれるビット数であり、シフトレジスタによる解決策のごとき1直列クロックサイクル当たり1つのビットを与えるのではなくて、PRGのNビット(Nは前記のごときWと等価である)のシリーズは、各並列クロックサイクルに対して発信される。k=N-iと置くと式(8)は、

$$Z^{\cdots k}=Z^{n}\sum_j B_{kj} Z^{j}=Z^{n} R_k \quad (9)$$

〔式中、$R_k=\Sigma_k B_k Z^{j}$は、行列のk行(または(N-i行)の係数をB_{kj}とする(N-1)次の多項式である〕と書ける。式(8)を、

$$Z^{0}=I=Z^{n}R_k \quad (10)$$

または

$$S_k(Z)=1+Z^{n}R_k=0 \quad (11)$$

で置き換える。

方程式$S_k(Z)=0$は、遷移行列の全ての行k($1<k<N$)に対して満足されるべき特性方程式である。

b)特性方程式の特徴

P(Z)のP個の根の連続累乗は方程式(4)の解である。信号Snはこれらの累乗の線形組合せであり、P(Z)の根の対称関数である。

従って直列シフトレジスタと同じ信号を生成するために、Sk(Z)の根はP(Z)の根を包含する必要があり、

$$S_k(Z)=P(Z)\ Q_k(Z) \quad (12)$$

〔式中、Sk及びPはZ_0項を有するので、Qk(Z)は少なくともかかる項を有するべき多項式である。〕である。この逆は成立するであろうか。

もし式(11):$S_k(Z)=P(Z)Q_k(Z)$であれば、$S_k(Z)=0$は、P(Z)の根に対してのみでなくQk(Z)の根に対しても真となる。ここで寄生根(parasitic root)として公知の問題が生じるが、「良いシード」(即ち良いPRGの一部分)を選択することにより、かかる寄生根の導入を回避することができる。これは、Sn,Sn-1等がラッチ内に包含され且つPRGの一部分であるならばSn+1は同じPRGのビットであるし、もしSn+1,Sn,Sn-1…がPRGの一部分であるならばSn+2はPRGのビットである等、Snの連続する値を考慮することにより証明され得る。

全ての多項式Sk(Z)は、Z^0から出発してP(Z)の倍数であらねばならない。このような多項式は、Z^nと$Z^{n\cdots n-1}$との間にのみ他の項(Pkの項)を有し、逆に、かかる多項式は並列発生機には都合が良い。kの同じ値に対してSk(Z)の幾つかの等価の式が存在し得る。

Skの異なる式が存在し得、例えば以下の2つの多項式:

$$S_{k1}(Z)=1+Z^{n}R_{k1}=P(Z)\ Q_{k1}(Z) \quad (13)$$

$$S_{k2}(Z)=1+Z^{n}R_{k2}=P(Z)\ Q_{k2}(Z) \quad (14)$$

は有効と言える。

唯一の条件は、Pk1及びPk2の次数がいずれもNより小さいことである。

上記2つの式を減算することにより、

$$Sk1-Sk2=Z^{n}(Rk1-Rk2)=P(Z)(Qk1-Qk2) \quad (15)$$

を得る。

第1に、多項式(Sk1−Sk2)はP(Z)で整除され、Sk1及びSk2は「P(Z)を法として合同」であると見なされる。これは、Sk2がP(Z)を基準としてSk1の項を置換することにより得られることを意味する。

例えばもしP(Z)=1+Z⁶+Z⁷であれば、Sk1において、1+Z⁶+Z⁷=0であるならばZ+Z⁷+Z⁸=0であるので、ZをZ⁷+Z⁸で置き換えることができる。

第2に、ゼロ根をもたない素数であるP(Z)はZⁿで整除されず、(Qk1−Qk2)はZⁿで整除される。(Rk1−Rk2)もまたP(Z)の倍数であり、Rk1及びRk2はP(Z)を法として合同である。

従って、行列の同じ行に対して等価の特性多項式Sk(Z)を与える(次数がNより小さい)「P(Z)を法として合同」の幾つかの多項式Rk(Z)が存在し得る。

第6図は幾つかの問題点を示す。

1) 考慮すべき2つの座標系が存在すること:1つは多項式Skに対するものであり、1つは多項式Rkに対するものである。これらは平行四辺形の内側または辺上になくてはならず(k=1〜k=8)、Rkの係数1が許容される位置を「/」で記した。

2) 選択した実施例においてはP(Z)の2つの重要な倍数、即ち、

$1+X^{6}+X^{7}$ 及び $1+Z^{17}+Z^{18}$

が存在し、Skの非定数項は、上記倍数の全てが許容範囲内にあるならばそのうちの1つの非定数項とすることができる。

3) 幾つかの多項式Skは同一であり(例えばS1〜S6)、対応するRkは項の平行移動のみが異なる。

C)Bezoutの関係

$$Sk(Z)=1+Z^{n}\ Rk=P(Z)\ Qk(Z) \quad (16)$$

または

$$1=Z^{n}\ Rk+P(Z)\ Qk(Z) \quad (17)$$

BEZOUTの関係を認識する(後述の表10の「BEZOUTの関係」参照)。

P(Z)及びQ(Z)を2つの多項式とすれば、それらの最大公約数(GCD)または最大公因数(HCF)は、

$$HCF(P,Q)=A(Z)P(Z)+B(Z)Q(Z) \quad (18)$$

で表わされる。A及びBは(ユークリッドの互除法から誘導される)極めて単純なアルゴリズムを用いて見つけることができ、Bの次数はPの次数よりも小さい。

Zⁿ及びP(Z)は公因数を持たないので(Pは既約である)、それらのHCFは1であり、式(17)はBEZOUTの関係である。

全ての値のkに対して、その次数がP(Z)の次数Pよりも小さい多項式Rkを決定することができる。式(17)においてk=1をとると1=ZR+PQとなり且つPQの次数は少なくともPに等しいので、Rの次数は多くともpに等しく、唯一の可能性はRの次数=p-1であり、Qの次数は0である。即ち少なくとも1つの多項式Rkはp個の項を有しており、遷移行列は少なくともp個の列を有する必要がある。

従って、

1) Nは少なくともpに等しくなければならない(N(ここではNと同じ)とpとの間に見られる関係説明参照)、

2) Pより大きいかまたはPに等しいNに対して、かかる問題に対する少なくとも1つの解がある。

この解は一般に、典型的には最小のXOR回路を求めるが故に最適ではない。しかしN>p-1であるならば、更に次数がNより小さいという条件でBEZOUTの関係によって与えられるものを用いてP(Z)を法として合同な多項式Rkを探すことにより解を向上する方法が存在する。

d)「発見的方法(Heuristic Solution)」

発見的方法は、2つ、3つまたはそれ以上の非定

数項を有するP(Z)の倍数を系統的に探索することからなる。行kに対して2つの係数の解が見付かったならば、それをk+1,…に対して出来る限り使用する。3つ以上の係数が必要であるならば、それらを行kに対してのみ使用し、次の行はより少ない係数を許容することが望まれるので、2つの係数を再度用いて開始する。テストするためにはP(Z)で除算し、余りがゼロ多項式か調べる。これは、高い値のp及びNに対しては計算時間が膨張することがあるが、いずれにせよ最速解を導くことができる(幾つかの解が存在する場合がある)。少なくとも現時点では最初に見付かった解を採用する。

Sk多項式をテストする別の方法は、多項式が所与の特性多項式によって生成された疑似乱数列を生成し得るか直接に検証することである。この方法はプログラム「GSPA-E」(表11及びサブルーチンPOLYANCOEFXのTEST部分参照)において使用されている。

目的に応じて勿論他の方法も可能である。例えば、条件を満足し得る多項式Skのテーブルを計算し、そのなかから選択して行列を構築することもより優れているであろう。

勿論、$(1+Z^6+Z^7)^2$のような倍数も良いことは明らかである。

まとめ

問題は、最小の係数を有するPの倍数を認知し、そのなかから、その非定数項がPk多項式の範囲内にあるものを選択することである。

e)プログラム

プログラムは4つの部分を含む。

1)初期化と、データ、即ち、

多項式P(Z)の次数pと、

A0及びAp(これらは常に1である)以外のPの係数とラッチのビット数Nと

の入力を行なう。更に、

Pが「良い」多項式であることを確認するため、

並列システムのテストのための良い「シード」を準備するため、及び

並列システムをテストする基準を設けるために、

P(Z)に対応するPRG配列(Seq1)を生成する。

2)行列の要素を計算する。

3)結果を印刷する。

行列の係数テーブル

Nが32以下である場合には行列の図

4)検証する。

配列(Seq2)を生成し、Seq1と比較する。

上記プログラムと一緒にサブルーチンファイルを使用する。サブルーチンファイルは全て本発明の目的に必要な2を法とする代数における全ての演算を包含する。

テーブル12は、GPSA-Eプログラムの実行による端末リストサンプルである。

テーブル13は、SONET多項式$(1+Z^6+Z^7)$の8、16、24、32及び64ビット並列ワード幅に対する幾つかのプリントアウトを包含する。

f)参考文献

Error Correcting Codes.

W.Wesley Peterson and E.J.Weldon (MIT Press)

Error Correction Coding for Digital Communications.

G.C Clark and J.B.Cain(PLENUM)

Shift Register Sequences

Solomon W.Golomb(Holden-Day,Inc.)

Sequences Pseudo-Aleatoires

Georges C.Roger(Laboratories de Marcoussis,Note Interne)

State Variables for Engineers(John Wiley & Sons,Inc.)

P.Derusso,R.Roy and C.Close,pp158-186

表10

「Bezoutの関係」

2つの整数a及びb、または2つの多項式の最大公因数(HCF)を求めたい場合、そのアルゴリズムはいずれに対しても同様である。

まずaをbで除算する。即ち、

$a = Q0\ b + R1 \qquad 0 \leq R1 \leq b$

とする。a及びbはHCFで割り切れ、更にR1もHCFで割り切れる。次にbをR1で除算する等を繰り返す。即ち、

$b = Q1\ R1 + R2 \qquad Q \leq R2 \leq R1$

$R1 = Q2\ R2 + R3 \qquad Q \leq R3 \leq R2$

.....

Rは次第により小さくなり、従って、

$Rn-2 = Qn-1\ Rn-1 + Rn$

--- (RnはHCFである)

$Rn-1 = Qn\ Rn + Rn+1 \qquad Rn+1 = 0$

Rn-1はRnで整除され、Rn-2もRnで整除され、・・・、a及びbもRnで整除される。

従ってRnはa及びbのHCFであり、これがユークリッドの互除法である。

次に連続する余りの数列：

$R1 = a - Q0\ b \quad A1\ a + B1\ b$ 但し$A1 = 1$及び$B1 = -Q0$

$R2 = b - Q1\ R1 \quad A2\ a + B2\ b \qquad A2 = -Q1\ A1$及び$B2 = -Q1\ B1$

$R3 = R1 - Q2\ R2 \quad A3\ a + B3\ b$

$A3 = A1 - Q2\ A2$及び$B3 = B1 - Q2\ B2$

.....

$Rn = An\ a + Bn$

但し$An = An-2 - Qn-1\ An-1$及び$Bn = Bn-2 - Qn-1\ Bn-1$

故に、An及びBnはA1及びB1から得られる。$A-1 = 1$、$B-1 = 0$(-1は添字である)且つ$A0 = 0$及び$B0 = 1$とすると、Rn、An及びBnを与えるアルゴリズムは、添字1から出発して単純に実行される。

もしRnがHCFであるならば、「Bezoutの関係」:

$$HCF(a,b) = A\ a + A\ b$$

が得られる。

aとbとが互いに素であるならば、$Rn = HCF(a,b) = 1$である。a及びbが多項式であるならばRnは定数であり、係数がF(0,1)に含まれる場合には1に等しい定数である。そして、

$$1 = a(Z)\ A(Z) + b(Z)\ B(Z)$$

の関係が得られる。

Anの次数は積Q1·Q2・・・・Qn-1の次数に等しく、Bnの次数は積Q0·Q1・・・・Qn-1の次数に等しいことは明らかであるし、計算することもできる。更に、·Aを多項式Aの次数を意味するとすると、

$\cdot Q0 = \cdot a - \cdot b$

$\cdot Q2 = \cdot b - \cdot R1$

$\cdot Q2 = \cdot R1 - \cdot R2$

.....

$\cdot Qn-1 = \cdot Rn-2 - \cdot Rn-1$

$\cdot Qn = \cdot Rn-1 - \cdot Rn$

となり、従って$\cdot(Q0 \cdot Q1 \cdot Q2 \cdots Qn-1) = \cdot a = \cdot Rn-1$である。

aとbとが互いに素であると仮定すると、Rnはそれらの HCFであり、$\cdot Rn = 0$及び$\cdot Rn-1$は少なくとも1である。従って$\cdot Bn$は$\cdot a$よりも小さく、同様の理由で$\cdot An$は$\cdot b$よりも小さい。

(表10終わり)

```
C  '   INPUT 0 TO INDICATE THE END OF THE OPERATION.',/)
10      CONTINUE
        WRITE (IUNIT2,7)
7       FORMAT (' RANK OF A COEFFICIENT EQUAL TO 1 ? ',$)
        READ (IUNIT),8)II
8       FORMAT (I)
        IF (II.EQ.0) GO TO 25  ! MEANS THE END OF THE INPUT
        P2(II+1)-1       !A0 IS IN P2(1), A1 IN P2(2)...
        GO TO 10         !NEXT COEF. TO INPUT
25      CONTINUE         !INPUT TERMINATED
C ******************
C ********** THE CHARACTERISTIC POLYNOMIAL HAS BEEN GIVEN. VERIFICATION!
C
        WRITE (IUNIT2,9500)
9500    FORMAT (/)
        CALL ECRIPOL (IUNIT2,'CHARACTERISTIC POLYNOMIAL:',DEG,P2,25,0)
C                           ** WRITES THE POLYNOMIAL **
        WRITE (IUNIT2,9)
9       FORMAT (/,' OK? !(RETURN-YES, IF NOT, INPUT: N ) ',$)
        REP=' '
        READ (IUNIT),12)REP
12      FORMAT (A)
        IF ( REP(1:1).EQ.'N')GO TO 10   ! TO STARTING POINT
C ********** THE WISHED COEFFICIENTS WERE ENTERED. IS THE POLYNOMIAL CORRECT?
C ******** WE VERIFY THAT POLYNOMIAL P2 GENERATES A MAXIMUM P.N.S.
        CALL SEQUENCE (IUNIT2,DEG,P,P2,A,SEQ1,INDIC)    !COMPUTES THE P.N.S.
        IF (INDIC.EQ.1) THEN
        WRITE (6,19)
19      FORMAT (/,'   THE P.N.S. IS NOT MAXIMUM !!!')
        GO TO 10         ! TO STARTING POINT
        END IF
C ******** THE CHARACTERISTIC POLYNOMIAL IS GOOD ******
9000    CONTINUE
30      WRITE (IUNIT2,13)
13      FORMAT (/,' 2) NUMBER OF SIMULTANEOUS BITS  TO BE ISSUED
     C  (N)<P-1)? ',$)
        READ (IUNIT),14)N
        IF (N.LT.P) GO TO 30    ! N MUST BE AT LEAST EQUAL TO P
14      FORMAT (I)
        WRITE (IUNIT2,9)         !OR?
        REP=' '
        READ (IUNIT),12)REP
        IF ( REP(1:1).EQ.'N')GO TO 30
C ************************************
C       END OF PARAMETERS INPUT.
C ************************************
C ************************************
C
        WRITE (IUNIT2,9500)
        WRITE (IUNIT2,203)
203     FORMAT (X,'*********************************')
        WRITE (IUNIT4,205) P
        WRITE (IUNIT2,205) P
205     FORMAT (' DEGREE OF CHARACTERISTIC POLYNOMIAL : ',I2,/)
        CALL ECRIPOL (IUNIT4,' CHARACTERISTIC POLYNOMIAL'
     C,DEG,P2,25,0)
        CALL ECRIPOL (IUNIT2,' CHARACTERISTIC POLYNOMIAL'
     C,DEG,P2,25,0)
        WRITE (IUNIT4,17)N
        WRITE (IUNIT2,17)N
17      FORMAT (/,' NB OF SIMULTANEOUS BITS: ',I3,/)
C ************************************************
C *** PARAMETERS ARE NOW SUMMARIZED ON THE SCREEN AND PUT IN THE FILE.
C ************************************************
C************************************************
C *** END OF INITIALISATIONS
```



```
C ******** PROGRAM GPSA E ************
C ******** COMPUTATION OF THE TRANSITION MATRIX OF A  PARALLEL  GENERATOR ***
C ************ OF PSEUDORANDOM SEQUENCE ****************************
C ***********(G.ROGER.LABORATOIRES DE MARCOUSSIS.OCTOBER 9 1988)******
C .                FORTRAN-77 VAX VMS
C*** DECLARATIONS
C
        CHARACTER*1 NOM,REP
        CHARACTER*4 NOM4
        CHARACTER*75 LIGNE
        INTEGER DEG
        PARAMETER (DEG=130,NBRN=64,NBBITS=10000) !NBBITS POUR DEGRE(-12
C **  POLYNOMIALS LIMITED TO DEGREE 128.PARAMETERS MAY BE AJUSTED AS NEEDED.
C **  NBBITS IS AT LEAST TWICE THE NUMBER OF BITS OF THE P.N.S.
C **  P IS THE DEGREE OF THE CHARACTERISTIC POLYNOMIAL GIVEN FOR THE SERIES
C **  GENERATOR. P2 IS THE CHARACTERISTIC POLYNOMIAL ITSELF.
        INTEGER P,P1(DEG),P2(DEG),Q(DEG),R(DEG)
        INTEGER AN (NBRN,DEG)    ! THE TRANSITION MATRIX
        INTEGER A(DEG)
        INTEGER B(DEG)
        INTEGER SEQ1 (NBBITS),SEQ2(NBBITS)
C **  SEQ1 AND SEQ2 ARE TWO P.N.S.
C ***************************
C **** INITIALISATIONS AND INPUT OF PARAMETERS *******
C ****************************************
        !!DEG=DEG
        IUNIT1-6         !SECONDARY RESULTS  (FOR VERIFICATIONS)
        IUNIT2-6         !THE SCREEN
        IUNIT3-5         !THE KEABOARD
        IUNIT4-10        !IMPORTANT RESULTS (ON THE FILE GPSA.DAT)
        OPEN (UNIT-10,FILE-'GPSA.DAT',STATUS-'NEW')
        WRITE (IUNIT2,301)
        WRITE (IUNIT4,301)
301     FORMAT (//,'  PARALLEL PSEUDONOISE SEQUENCIES GENERATOR.',/,
     C  /,'       COMPUTATION OF THE TRANSITION MATRIX.( PROGRAM GPSA)',/,
     C  '              GEORGES ROGER L.D.M. 9/11/88',//)
C ***********************************
C ************** INPUT OF PARAMETERS ********
C ***********************************
10      WRITE (IUNIT2,1)
1       FORMAT (//,' 1) DEGREE OF THE CHARACTERISTIC POLYNOMIAL
     C  (1<P<13)? ',$)
        READ (IUNIT),2)P
2       FORMAT (I)
        IF (P.LT.2)THEN
        WRITE (IUNIT2,3)
3       FORMAT (' DEGREE TOO SMALL!')
        GO TO 10
        END IF
        IF (P.GT.12)THEN
        WRITE (IUNIT2,4)
4       FORMAT (' DEGREE TOO BIG!')
        GO TO 10
        END IF
        WRITE (IUNIT2,201)P
201     FORMAT (' DEGREE OF CHARACTERISTIC POLYNOMIAL: ',I2)
        CALL PMUL (DEG,P2)       ! CREATES THE CHARACT. POLYN. WITH
        P2(P+1)-1                ! LAST AND
        P2(1)-1                  ! FIRST COEFFICIENTS (ALWAYS EQUAL TO 1)
        WRITE (IUNIT2,5)         ! THEN, ASKS FOR OTHER COEFFICIENTS:
5       FORMAT (/,'     INPUT OF THE CHARACTERISTIC POLYNOMIAL.',//,
     C  '  POLYNOMIALSS ARE WRITTEN AS:',/,
     C  '  X0 + A1 X1 + A2 X2 + .... + AP-1 XP-1 +XP',/,
     C  '  PLEASE GIVE THE RANK OF COEFFICIENTS'A1 TO AP-1 EQUAL TO 1,
     CONE AFTER THE OTHER.',/,
     C  '  (THOSE OF DEGREE 0 AND P ARE  EQUAL TO  1 ALREADY)',/,
```

```
      IF (N.GT.32) THEN
      WRITE (IUNIT4,611)
611   FORMAT (//,' MATRIX TOO BIG.UNABLE TO WRITE IT... ',//)
      GO TO 600
      END IF
607   FORMAT (X,A)
      WRITE (IUNIT4,9500)
      WRITE (IUNIT4,201)
      WRITE (IUNIT4,205) P
      CALL ECRIPOL (IUNIT4,' CHARACTERISTIC POLYNOMIAL ',
     COEG,P1,25,0)
      WRITE (IUNIT4,17)N
      WRITE (IUNIT4,203)
      WRITE (IUNIT4,9500)
      WRITE (IUNIT4,601)
601   FORMAT (//,10X,'MATRIX:',/)
      IF (N.LE.24)KE=N
      IF (N.GT.24)KE=2
      WRITE (IUNIT4,605)((JJ-1)/10,JJ=1,N)
605   FORMAT (5X,<N>(I<KE-1>,X))
      WRITE (IUNIT4,603)(JJ-1-10*((JJ-1)/10),JJ=1,N)
603   FORMAT (5X,<N>(I<KE>),/)
      DO 600 II=1,N
      LIGNE=' '
      DO 610 JJ=1,N
      JJ1=KE*JJ
      IF (RN(II,JJ).EQ.0)LIGNE (JJ1:JJ1)='-'
      IF (RN(II,JJ).EQ.1)LIGNE (JJ1:JJ1)='+'
610   CONTINUE
      WRITE (IUNIT4,617)II-1,LIGNE
617   FORMAT (X,I2,2X,A75)
600   CONTINUE
      WRITE (IUNIT4,9500)
C ********* THE MATRIX IS WRITTEN
C
C *************** VERIFICATION ***************************
      IPERMAX=2**P-1
      NOMBREDEBITS=2*IPERMAX+100
C ************** VERIFICATION OF THE PARALLEL GENERATOR  *******
C *** WE USE POLYNOMIAL 'A' FOR STATE  N AND 'B' FOR STATE N+1
C **  WE LOAD THE GENERATOR WITH THE FIRST N VALUES OF SEQ1
      DO 5000  II=1,N
5000  A(N-II+1)=SEQ1(II)      ! TAKE CARE OF THE TIME INCREASING DIRECTION!
C ***   WE NOW COMPUTE THE VALUES TO PUT INTO B:
C
      DO 5500 M=1,NOMBREDEBITS/N
      DO 5100 II=1,N
      B(II)=0
      DO 5200 JJ=1,N
      IF (RN(II,JJ).EQ.1)  B(II)=B(II).XOR.A(JJ)
5200  CONTINUE
5100  CONTINUE
C **  WE PUT THE VALUES OF B IN SEQ2:
      DO 5300 II=1,N
      SEQ2((M-1)*N+II)=A(N-II+1)
5300  CONTINUE
C **  WE PUT THE VALUES OF B IN 'A' FOR THE FOLLOWING STATE:
      DO 5400 II=1,N
5400  A(II)=B(II)
5500  CONTINUE
C ************** END OF THE COMPUTATION OF SEQ2, OBTAINED WITH THE PARALLEL
C ** GENERATOR, AND COMPARISION WITH SEQ1, OBTAINED WITH A SERIES GENERATOR.
      WRITE (IUNIT1,470)(SEQ1(KK),KK=1,72)
470   FORMAT (' SEQ1:',X,72I1)
      WRITE (IUNIT1,570)(SEQ2(KK),KK=1,72)
570   FORMAT (' SEQ2:',X,72I1)
```

```
C ***************************************************************
C***************************************************************
      WRITE (IUNIT2,207)
207   FORMAT (/,X,' COMPUTING .....')
      WRITE (IUNIT2,203)
C ** LOOKING FOR THE SOLUTION **************
      KK=1     !FIRST LINE AT THE BOTTOM OF THE MATRIX
1000  CONTINUE
      IF (KK.GT.N)GO TO 2900  ! THE END
      IDEP=KK
      NBC=2    !NB DE COEFF DE ZK KK AUTRES QUE 10
C  ** WE LOOK FOR A POLYNOMIAL P1 MULTIPLE OF P2 AND HAVING ONLY
C  ** NBC-TWO COEFFICIENTS BETWEEN KK AND KK+N-1.SUCH A POLYNOMIAL IS A
C  ** GENERATOR OF THE P.N.S. SEQ1, ALREADY COMPUTED.
C1    WRITE (IUNIT1,2001)KK,N+KK-1,NBC
2001  FORMAT (' KK= ',I3,'   N+KK-1= ',I3,'  NBC= ',I3)
      CALL POLYANCOEFX (DEG,P1,KK,N+KK-1,NBC,P2,INDICP,SEQ1)
C1    IF (INDICP.EQ.1)WRITE (IUNIT1,2003)INDICP        ! FOR TEST OR DEBUGGING
2003  FORMAT (' INDICP= ',I3)
C ***************************************CAS OU NBC =2
      IF (INDICP.EQ.1) THEN   !SUCCESS
      P1(1)=0                 !TO LOOK FOR THE DEGREE OF THE 1ST COEF OTHER THAN
C                             ! 10, ALWAYS EQUAL TO 1
      CALL DEGDEB (P1,DEG,IDEGDEB,INDIC) !IDEGDEB=DEGREE OF THE FIRST COEFF
C1    WRITE (IUNIT1,2005)IDEGDEB
2005  FORMAT (' IDEGDEB= ',I3)
C ***********  P1 MAY BE USED UNTIL  N-IDEGDEB **********
C ******* POLYNOMIAL P1 IS COPIED IN THE MATRIX RN WITH THE CORRECT SHIFT
      DO 2100 KK1=KK,IDEGDEB
      IF (KK1.GT.N) GO TO 2900        ! FINISHED
      DO 2200 LL=1,N  ! P1 COPIED IN RN
      RN (N-KK1+1,LL)=P1(LL+KK1)
2200  CONTINUE
      NOM4='RN  '
      WRITE (NOM4(3:4),'(I2.2)')N-KK1+1
      CALL ECRIPOLTAB ((IDEG,NBRN,IUNIT4,NOM4,DEG,RN,N-KK1+1,70,0)
2100  CONTINUE
      KK=IDEGDEB+1
      GO TO 2000
      END IF,          !IF INDICP=0, NOSUCCESS, WE TRY WITH 3, 4, ETC. COEFF.
C ***************************************************
2300  CONTINUE
      NBC=NBC+1
C1    WRITE (IUNIT1,2001)KK,N+KK-1,NBC
      CALL POLYANCOEFX (DEG,P1,KK,N+KK-1,NBC,P2,INDICP,SEQ1)
C1    IF (INDICP.EQ.1)WRITE (IUNIT1,2003)INDICP
      IF (INDICP.EQ.1) THEN   !SUCCES
      DO 2250 LL=1,N
      RN (N-KK+1,LL)=P1(LL+KK)
2250  CONTINUE                  ! P1 PUT IN RN
      NOM4='RN  '
      WRITE (NOM4(3:4),'(I2.2)')N-KK1+1
      CALL ECRIPOLTAB ((IDEG,NBRN,IUNIT4,NOM4,DEG,RN,N-KK+1,70,0).
      KK=KK+1
      GO TO 2000       ! NEXT LINE OF THE MATRIX
      END IF           ! OR NOSUCCESS
      GO TO 2300       ! ONE MORE COEFF.
C
2900  CONTINUE
3000  CONTINUE
C ******* THE END ....
C ***********************************
C ************** END OF THE COMPUTATION ************
C ***********************************************
C ***** WRITING THE MATRIX IN THE FILE ************
      WRITE (IUNIT4,607)CHAR (12) ! FORM FEED
```

```
C ** DES ENTIERS MODULO 2
C
C   LISTE DES SOUSROUTINES DE TRAITEMENT DE POLYNOMES A COEFFICIENTS
C   SUR LE CORPS (0,1)
C
C       LES POLYNOMES SONT ORDONNES PAR DEGRE CROISSANT
C
C       COPIE (DEG,P1,P2)                ;RECOPIE P1 DANS P2
C       DEGRE (POL,DEG,DEGM,INDIC)       ;DONNE LE DEGRE DE POL
C       DEGDEB (POL,DEG,DEGD,INDIC       ;DONNE LE PREMIER TERME NON NUL
C       PNUL (DEG,P)                     ;CREE LE POLYNOME P IDENTIQUEMENT NUL
C       PUNIT (DEG,P)                    ;CREE LE POLYNOME EGAL A 1
C       ECRIPOL (IUNIT,NOM,DEG,P,DEGMAX,INDIC)
C                                        ;ECRIT LE POLYNOME P SUR L'UNITE IUNIT,
C                                         PRECEDE D'UN TITRE 'NOM', JUSQ'A DEGMAX
C                                         SI INDIC=0, LES TERMES SONT GROUPES
C                                         SI INDIC=1, LA PLACE DES TERMES NULS
C                                         EST REMPLACEE PAR DES BLANCS.
C       ECRIPOLTAB (IIDEG,NBRN,IUNIT,NOM,DEG,P,I,DEGMAX,INDIC)
C                                        ;COMME ECRIPOL, MAIS POUR UN TABLEAU
C                                         DE POLYNOMES, DONT ON ECRIT LA LIGNE I.
C       POLYANCOEFX                      ;CHERCHE LES MULTIPLES D'UN POLYNOME
C                                        ; QUI ONT UN NOMBRE DONNE DE COEFFICIENTS
C**************************************************
C ***  REMARQUES GENERALES. REMARKS.
C  LES POLYNOMES SONT TOUS ECRITS DANS UN VECTEUR A DEG POSITIONS
C  EN PARTANT DU DEGRE NUL.
C  POLYNOMIALS ARE WRITTEN IN AN ARRAY, STARTING FROM 0 DEGREE TERM.
C  LE DEGRE MAXIMAL TRAITABLE EST DONC DEG-1.ATTENTION AUX DEBORDEMENTS!!!
C  LES POLYNOMES AUXILIAIRES DONT LES DEGRES NE SONT PAS PASSES EN ARGUMENT
C  SONT DIMENSIONNES A 256.
C
C ******************************
C******************************
      SUBROUTINE DEGRE (POL,DEGM,DEG,INDIC)
C  DONNE LE DEGRE DU POLYNOME POL ECRIT DANS DEGM CELLULES
C ** DEGM-DEGRE MAX+1,DEG-DEGRE DU POLYNOME
C ** LOOKS FOR THE DEGREE OF THE POLYN.INDIC-1 IF THE POLYN. IS NULL
      INTEGER DEGM,DEG ,POL(1)
      INDIC=0
      DO 200 II=DEGM,1,-1
      IF (POL(II).EQ.1)GO TO 210
200   CONTINUE
      IF (POL(1).EQ.0) THEN
      DEG=-1              !INDICATION DE DEGRE NUL
C     WRITE (6,1)
1     FORMAT (X,'DEGRE: POLYNOME IDENTIQUEMENT NUL!')
      INDIC=1
C     GO TO 10000
      END IF
210   DEG=II-1
10000 CONTINUE
      RETURN
      END
C ************************************
C************************************
      SUBROUTINE DEGDEB (POL,DEGM,DEGB,INDIC)
C ** DEGM-DEGREMAX,DEGB-DEGRE DU 1ER TERME DU POLYNOME
C ** INDIC=1 SI POLYNOME NUL
C ** LOOKS FOR THE FIRST TERM OF THE POLYN.
      INTEGER DEGM,DEGB ,POL(1)
      INDIC=0
      DO 200 II=1,DEGM
      IF (POL(II).EQ.1)GO TO 210
200   CONTINUE
      IF (POL(DEGM).EQ.0) THEN
      DO 6000 II=1,2*IPERMAX
      IF (SEQ1(II).NE.SEQ2(II)) GO TO 6100
6000  CONTINUE
      WRITE (IUNIT4,6001)
6001  FORMAT (//,'   VERIFICATION O.K.  !!!')
      GO TO 10000
6100  WRITE (IUNIT4,6201)
6201  FORMAT (' THE P.M.S. ARE DIFFERENT. !!! SOMETHING WRONG.')
10000 CONTINUE
      WRITE (IUNIT2,209)
209   FORMAT (//,'   JOB TERMINATED.RESULTS IN GFSA.DAT',//)
      END
C**************************************************************
      SUBROUTINE SEQUENCE (IUNIT,DEG,P,P2,A,SEQ1,INDIC)
C  ** ON GENERE LA SEQUENCE QUE FOURNIRAIT LE SHIFT REGISTER A
C     DE P BASCULES ET DE POLYNOME CARACTERISTIQUE P2.
C     DEUX PERIODES DE LA SEQUENCE MAXIMALE SONT RANGEES DANS SEQ1.
C     SI LA PERIODE N'EST PAS MAXIMALE, L'INDICATEUR EST MIS A 1
C       LES MESSAGES SONT ECRITS SUR IUNIT
C
C ** GENERATES THE SEQUENCE SEQ1 FURNISHED BY THE SHIFT REGISTER A,
C ** OF P STAGES AND CHARACTERISTIC POLYNOMIAL P2
C ** TWO PERIODS ARE COMPUTED. IF THE P.M.S. IS NOT MAXIMUM, INDIC=1
C ** MESSAGES ARE WRITTEN ON IUNIT1
      INTEGER DEG,P
      INTEGER  P2(1),A(1),SEQ1(1)
      INDIC=0
C ** LOADING 'A' WITH THE SEED
      CALL PNUL(DEG,A)
      DO 4100 II=2,P+1
4100  A(II)=1           ! DONE
C ********* ALGORITHM *************
      IPERMAX=2**P-1 ! PERIOD
      NOMBREDEBITS=2*IPERMAX+100       ! NOMBREDEBITS > 2 * IPERMAX
      DO 4500 MM=1,NOMBREDEBITS        ! MORE THAN TWO PERIODS.
C *** ON CALCULE L'ELEMENT QUE VA ENTRER DANS LE REGISTRE.C'EST A(1)
      A(1)=A(P+1)      ! ALWAYS CONNECTED
      DO 4200 II=2,P   ! IF A(II) CONNECTED...
      IF (P2(II).EQ.1) A(1)=A(1).XOR.A(II)
4200  CONTINUE
C ** COLLECTING THE ISSUED BIT AND SHIFTING
      SEQ1(MM)=A(P+1) ! COLLECTING
      DO 4300 II=P+1,2,-1      ! SHIFTING
4300  A(II)=A(II-1)
4500  CONTINUE
C ******* ON VERIFIE LA PERIODE. PERIOD VERIFICATION.
      DO 4600 IPER=1,IPERMAX+1
      DO 4650 II=1,IPER+10
      IF (SEQ1(II+IPER).NE.SEQ1(II)) GO TO 4600
4650  CONTINUE
      GO TO 4700
4600  CONTINUE
4700  CONTINUE
      WRITE (IUNIT,4701)IPER,IPERMAX
4701  FORMAT (//,' PERIOD OF SEQ1: ',
     C  I6,' MAX PERIOD : ',I6)
      WRITE (IUNIT,4703)(SEQ1(KK),KK=1,72)
4703  FORMAT (' SEQ1:',X,72I1)
      IF(IPER.NE.IPERMAX)INDIC=1
      RETURN
      END
C **************************************************************
C **************************************************************
C ******** POLY_2_OP.FOR ***
C **************************************************************
C ** BIBLIOTHEQUE D'OPERATIONS SUR LES POLYNOMES DONT LES COEFFICIENTS SONT
```

```
      IF ((II-1).GT.99.AND.(II-1).LT.1000)
     C  WRITE(LIGNE(IDEP+1:IDEP+3),'(I3)')II-1
120   IDEP=IDEP+4
      IF((IDEP+8).GT.80) THEN !2
      LIGNE(IDEP:IDEP+3)='....'
      WRITE (IUNIT,101)
101   FORMAT (X,' RESULT IS CUT')
      GO TO 1000
      END IF                  !2
      END IF                  !1
100   CONTINUE
1000  CONTINUE
C ***********************************************
      WRITE (IUNIT,103)LIGNE
103   FORMAT (X,A80)
      RETURN
      END
C ***********************************************
      SUBROUTINE ECRIPOLTAB (IIDEG,NBRN,IUNIT,NOM,DEG,P,I,DEGMAX,INDIC)
C *** ECRIT LE POLYNOME DE RANG I DU TABLEAU DE POLYNOMES P
C *** JUSQU'AU DEGRE DEG MAX
C ******** WRITES A POLYNOMIAL OF RANK I, TAKEN IN A TABLE OF POLYNOMIALS
      CHARACTER*(*)NOM
      CHARACTER*80    LIGNE
      INTEGER DEG,DEGMAX,DEGMAX1
      INTEGER P(NBRN,IIDEG),A (300)
      DO 10 II=1,DEG
10    A(II)=P(I,II)
      CALL ECRIPOL (IUNIT,NOM,DEG,A,DEGMAX,INDIC)
      RETURN
      END
C ***************************************************************
C ***************************************************************
      SUBROUTINE POLYANCOEFX (DEG,P2,DEGDEB,DEGMAX,N,P,INDIC,SEQ)
C ** TROUVE, S'IL EXISTE, UN POLYNOME P2, DE DEGRE MAXIMAL DEGMAX,
C ** COMMENCANT PAR LE TERME DE DEGRE DEGDEB, CONTENANT
C ** EXACTEMENT N TERMES NON NULS AUTRES QUE LE TERME CONSTANT, TOUJOURS
C ** SUPPOSE EGAL A 1, ET QUI EST UN MULTIPLE DU POLYNOME P.LE POLYNOME Q
C                                                  n   n
C ** EST LE MULTIPLICATEUR.EXEMPLE, SI N=2:  1+X + X  = P Q   P,n, A TROUVER.
C*********
C************ FINDS, IF IT EXISTS, A POLYNOMIAL P2, THE NON CONSTANT TERMS OF
C ** WHICH ARE BETWEEN DEGDEB AND DEGMAX (INCLUDED), HAVING EXACTLY N NON
C ** CONSTANT TERMS AND MULTIPLE OF P. IN FACT, WE USE THE P.N.S. SEQ GENERATED
C ** BY P TO TEST THAT P2 IS A GENERATOR OF SEQ.
C                                  n   n
C ** EXAMPLE, IF N=2 :  1+X + X  = P Q   P,n, to be found.
      INTEGER DEG,DEGDEB,DEGMAX,N
      INTEGER P (1),P2(1),SEQ(1)
      INTEGER Q (256),Q(256)
      INTEGER INDICE (6)      !POSITIONS OF THE INDICES.
      CALL DEGRE (P,DEG,IDEGP,INDICD)
      NMAX=2**IDEGP-1         !PERIOD OF THE P.N.S.
      INDIC=0                 !INDIC=1 EN CAS DE SUCCES.
      IDEP=DEGDEB             !DEGDEB=0 , THE FIRST POSITION WILL BE 0+1
      DO 10 II=1,N            !THE N TERMS ARE POSITIONNED BLOCKED TOGETHER, STARTING
C                             !FROM DEGDEB. THEY WILL BE MOVED AFTER EACH TEST.
      INDICE(II)=IDEP+1   ! INDICE(II) IS THE POSITION OF TERM II
10    CONTINUE            !
100   CONTINUE            !RETURN POINT
      CALL PUNIT (DEG,P2)     !CREATES THE UNITY POLYN.
      DO 120  II=1,N
      P2(INDICE(III)=1        !PUTS THE NON CONSTANT TERMS A THEIR PLACES
120   CONTINUE                !IN THE POLYNOMIAL
C ***********************************  TEST ************
C ***** LOOKS TO SEE IF THE POLYN. IS A GENERATOR OF SEQ.
C       WRITE (6,1)
1       FORMAT (X,'DEGDEB: POLYNOME IDENTIQUEMENT NUL!')
        INDIC=-1
C       GO TO 10000
        END IF
210     DEGB=II-1
10000   CONTINUE
        RETURN
        END
C ***************************************
C ***************************************
        SUBROUTINE COPIEX (DEG,P1,P2)
C ** RECOPIE P1 DANS P2 TERME A TERME.COPIES P1 INTO P2
        INTEGER DEG
        INTEGER P1(1),P2(1)
        DO 100 II=1,DEG
        P2(II)=P1(II)
100     CONTINUE
        RETURN
        END
C ***************************************
C ***************************************
        SUBROUTINE PNUL (DEG,P)
C ** CREE LE POLYNOME NUL DE DEGRE MAX DEG. CREATES THE NULL POLYNOMIAL
        INTEGER DEG
        INTEGER P(1)
        DO 100 II=1,DEG
100     P(II)=0
        RETURN
        END
C ***************************************
        SUBROUTINE PUNIT (DEG,P)
C ****** CREE LE POLYNOME DE DEGRE 0.CREATES THE CONSTANT POLYN. *******
        INTEGER DEG
        INTEGER P(1)
        CALL PNUL (DEG,P)
        P(1)=1
        RETURN
        END
C ***************************************
C ***************************************************
        SUBROUTINE ECRIPOL (IUNIT,NOM,DEG,P,DEGMAX,INDIC)
C **    SI INDIC=0,TERMES BLOQUES,SI INDIC=1,ESPACES DE 4 BLANCS POUR CHAQUE
C **      TERME NUL
C ** IUNIT:L'UNITE LOGIQUE SUR LAQUELLE ON ECRIT.
C ** WRITES A POLYNOMIAL P. IF INDIC=0, NULL TERMS ARE DISCARDED.IF INDIC=1,
C ** FOUR BLANKS ARE LEFT FOR EACH BLANK TERM. DEGMAX LIMITS THE NUMBER OF
C ** WRITTEN TERMS. NOM IS THE NAME OF THE POLYNOMIAL, WHICH MAY BE WRITTEN.
        CHARACTER*(*)NOM
        CHARACTER*80    LIGNE
        INTEGER DEG,DEGMAX,DEGMAX1
        INTEGER P(1)
        CALL DEGRE (P,DEG,DEGMAX1,INDIC2)
        L=LEN (NOM)
        LIGNE=' '
        LIGNE (1:L)=NOM
        LIGNE (L+1:L+3)=' :'
        IDEP=L+4
C ***************************************
        DO 100 II=1,DEGMAX
        IF (P(II).EQ.0.AND.INDIC.EQ.1) GO TO 120        !ON SAUTE 4 BLANCS
        IF (P(II).EQ.1) THEN    !1
        LIGNE(IDEP:IDEP)='1'
        IF ((II-1).LE.9)WRITE(LIGNE(IDEP+1:IDEP+1),'(I1)')II-1
        IF ((II-1).GT.9.AND.(II-1).LT.100)
     C  WRITE(LIGNE(IDEP+1:IDEP+2),'(I2)')II-1
```

表12

SAMPLE TERMINAL LISTING
-RUNNING PGM GPSA - E

```
$ RUN GPSA_E


 PARALLEL PSEUDONOISE SEQUENCIES GENERATOR.

     COMPUTATION OF THE TRANSITION MATRIX.( PROGRAM GPSA)
           GEORGES ROGER L.D.H. 9/11/88


1) DEGREE OF THE CHARACTERISTIC POLYNOMIAL   (1<P<13)? 7
DEGRE DU POLYNOME CARACTERISTIQUE:  7

    INPUT OF THE CHARACTERISTIC POLYNOMIAL.

 POLYNOMIALSS ARE WRITTEN AS:
 X0 + A1 X1 + A2 X2 + .... + AP-1 XP-1 +XP
 PLEASE GIVE THE RANK OF COEFFICIENTS A1 TO AP-1 EQUAL TO 1,  ONE AFTER THE OTHER

 (THOSE OF DEGREE 0 AND P ARE  EQUAL TO  1 ALREADY)
  INPUT 0 TO INDICATE THE END OF THE OPERATION.

RANK OF A COEFFICIENT EQUAL TO 1 ? 6
RANK OF A COEFFICIENT EQUAL TO 1 ? 0


CHARACTERISTIC POLYNOMIAL: : Z0  Z6  Z7

 K? :(RETURN=YES, IF NOT, INPUT: N ) Y


PERIODE TROUVEE POUR SEQ1:     127  PERIODE MAX:     127
 EQ1: 1111111000000100000110000101000111100100010111001110101001111101000011100

 ) NUMBER OF SIMULTANEOUS BITS     (N>P-1)? 8

 K? :(RETURN=YES, IF NOT, INPUT: N ) Y


 ********************************
DEGREE OF CHARACTERISTIC POLYNOMIAL :  7

 CHARACTERISTIC POLYNOMIAL : Z0  Z6  Z7

   OF SIMULTANEOUS BITS:   8


 COMPUTING .....
 ********************************
 EQ1: 1111111000000100000110000101000111100100010111001110101001111101000011100
 EQ2: 1111111000000100000110000101000111100100010111001110101001111101000011100


 JOB TERMINATED.RESULTS IN GPSA.DAT
```

```
      DO 2100 IECH=1+DEG  I,NMAX+DEGMAX        ! IECH-1, THE RANK OF ONE BIT
      ITEST=SEQ (IECH)                         ! OF THE P.N.S. SEQ
      DO 2000 II=1,N
      ITEST = ITEST.XOR.SEQ(IECH-INDICE(II)+1)
2000  CONTINUE
      IF (ITEST.NE.0) GO TO 2200       ! INTERRUPTS THE TEST AS SOON AS A
C                                      ! DISCREPANCY APPEARS.
2100  CONTINUE
      GO TO 1000       !SUCCESS!!!
C ** SINON, ON ESSAIE DE DECALER LES INDICES
C ** IF NO SUCCESS, THE TERMS OF THE POLYN. ARE SHIFTED
2200  CONTINUE
C ********************** END OF TEST ************************
      DO 200 II=1,N-1
      IF (INDICE(II).LT.INDICE(II+1)-1) THEN  !THERE IS ROOM TO SHIFT THE TERM
      INDICE (II)=INDICE(II)+1                !IT IS DONE
      DO 210 LL=1,II-1          ! PRECEEDING TERMS ARE PLACED AGAIN AT THE
210   INDICE(LL)=LL+IDEP        ! BEGINNING
      GO TO 100                          !TO THE TEST
      END IF
200   CONTINUE                  ! WHEN HERE, ALL TERMS ARE BLOCKED AGAINST THE
C                               ! NTH TERM, WHICH MUST BE SHIFTED
      INDICE(N)=INDICE(N)+1     ! TERM N IS SHIFTED
C ******* FIN
      IF (INDICE(N).GT.DEGMAX+1) THEN !STOP!,N GREATER THAN DEGMAX
      INDIC=0          !NO SUCCESS AT ALL.EXIT
      GO TO 10000
      END IF
C ************************
      DO 210 LL=1,N-1 !OTHER TERMS ARE PLACED AGAIN AT THE BEGINNING
      INDICE(LL)=IDEP+LL
210   CONTINUE
      GO TO 100        !TO THE RETURN POINT
1000  CONTINUE         !SUCCESS!!!!
      INDIC=1
10000 CONTINUE
      RETURN
      END
C*******************************************************
```

```
$ TY GPSA.DAT

PARALLEL PSEUDONOISE SEQUENCIES GENERATOR.

        COMPUTATION OF THE TRANSITION MATRIX.( PROGRAM GPSA)
                GEORGES ROGER L.D.H. 9/11/88

DEGREE OF CHARACTERISTIC POLYNOMIAL :  7

 CHARACTERISTIC POLYNOMIAL : Z0  Z6  Z7

NB OF SIMULTANEOUS BITS:  8

RN07 : Z5  Z6
RN06 : Z4  Z5
RN05 : Z3  Z4
RN04 : Z2  Z3
RN03 : Z1  Z2
RN02 : Z0  Z1
RN01 : Z5  Z7
RN00 : Z4  Z6
```

```
**********************************
DEGREE OF CHARACTERISTIC POLYNOMIAL :  7
 CHARACTERISTIC POLYNOMIAL : Z0  Z6  Z7
NB OF SIMULTANEOUS BITS:  8
**********************************

        MATRIX:

      0  0  0  0  0  0  0  0
      0  1  2  3  4  5  6  7
   0  -  -  -  -  -  +  -  +
   1  -  -  -  -  +  -  +  -
   2  -  -  -  +  -  +  -  -
   3  -  -  +  -  +  -  -  -
   4  -  +  -  +  -  -  -  -
   5  +  -  +  -  -  -  -  +
   6  -  +  -  -  -  -  +  -
   7  +  -  -  -  -  +  -  -

VERIFICATION O.K.  !!!
```

表13

```
PARALLEL PSEUDONOISE SEQUENCIES GENERATOR.

        COMPUTATION OF THE TRANSITION MATRIX.( PROGRAM GPSA)
                GEORGES ROGER L.D.H. 9/11/88

DEGREE OF CHARACTERISTIC POLYNOMIAL :  7

 CHARACTERISTIC POLYNOMIAL : Z0  Z6  Z7

NB OF SIMULTANEOUS BITS:  8

RN07 : Z5  Z6
RN06 : Z4  Z5
RN05 : Z3  Z4
RN04 : Z2  Z3
RN03 : Z1  Z2
RN02 : Z0  Z1
RN01 : Z5  Z7
RN00 : Z4  Z6
```

```
.....................................
DEGREE OF CHARACTERISTIC POLYNOMIAL :  7
 CHARACTERISTIC POLYNOMIAL  : Z0  Z6  Z7
NB OF SIMULTANEOUS BITS:  16
.....................................
```

MATRIX:

	0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	1 0	1 1	1 2	1 3	1 4	1 5
0	-	-	-	-	-	-	-	-	+	-	-	-	+	-	-	-
1	-	-	-	-	-	-	-	-	-	+	-	-	-	+	-	-
2	-	-	-	-	-	-	-	-	-	-	+	-	-	-	+	-
3	-	-	-	-	-	-	-	-	-	-	-	+	-	-	-	+
4	+	-	+	-	-	-	-	-	-	-	-	-	-	-	-	-
5	-	+	-	+	-	-	-	-	-	-	-	-	-	-	-	-
6	-	-	+	-	+	-	-	-	-	-	-	-	-	-	-	-
7	-	-	-	+	-	+	-	-	-	-	-	-	-	-	-	-
8	-	-	-	-	+	-	+	-	-	-	-	-	-	-	-	-
9	-	-	-	-	-	+	-	+	-	-	-	-	-	-	-	-
10	+	+	-	-	-	-	-	-	-	-	-	-	-	-	-	-
11	-	+	+	-	-	-	-	-	-	-	-	-	-	-	-	-
12	-	-	+	+	-	-	-	-	-	-	-	-	-	-	-	-
13	-	-	-	+	+	-	-	-	-	-	-	-	-	-	-	-
14	-	-	-	-	+	+	-	-	-	-	-	-	-	-	-	-
15	-	-	-	-	-	+	+	-	-	-	-	-	-	-	-	-

VERIFICATION O.K. !!!

PARALLEL PSEUDONOISE SEQUENCIES GENERATOR.

COMPUTATION OF THE TRANSITION MATRIX.(PROGRAM GPSA
GEORGES ROGER L.D.M. 9/11/88

DEGREE OF CHARACTERISTIC POLYNOMIAL : 7

CHARACTERISTIC POLYNOMIAL : Z0 Z6 Z7

NB OF SIMULTANEOUS BITS: 16

```
RN15 : Z5  Z6
RN14 : Z4  Z5
RN13 : Z3  Z4
RN12 : Z2  Z3
RN11 : Z1  Z2
RN10 : Z0  Z1
RN09 : Z5  Z7
RN08 : Z4  Z6
RN07 : Z3  Z5
RN06 : Z2  Z4
RN05 : Z1  Z3
RN04 : Z0  Z2
RN03 : Z11 Z15
RN02 : Z10 Z14
RN01 : Z9  Z13
RN00 : Z8  Z12
```

PARALLEL PSEUDONOISE SEQUENCIES GENERATOR.

COMPUTATION OF THE TRANSITION MATRIX.(PROGRAM GPSA)
GEORGES ROGER L.D.M. 9/11/88

DEGREE OF CHARACTERISTIC POLYNOMIAL : 7

CHARACTERISTIC POLYNOMIAL : Z0 Z6 Z7

NB OF SIMULTANEOUS BITS: 24

```
RN23 : Z5  Z6
RN22 : Z4  Z5
RN21 : Z3  Z4
RN20 : Z2  Z3
RN19 : Z1  Z2
RN18 : Z0  Z1
RN17 : Z5  Z7
RN16 : Z4  Z6
RN15 : Z3  Z5
RN14 : Z2  Z4
RN13 : Z1  Z3
RN12 : Z0  Z2
RN11 : Z11 Z15
RN10 : Z10 Z14
RN09 : Z9  Z13
RN08 : Z8  Z12
RN07 : Z7  Z11
RN06 : Z6  Z10
RN05 : Z5  Z9
RN04 : Z4  Z8
RN03 : Z3  Z7
RN02 : Z2  Z6
RN01 : Z1  Z5
RN00 : Z0  Z4
```

```
**********************************
DEGREE OF CHARACTERISTIC POLYNOMIAL :  7
 CHARACTERISTIC POLYNOMIAL  : Z0  Z6  Z7
NB OF SIMULTANEOUS BITS:  24
**********************************
```

MATRIX:

	0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	2 0	2 1	2 2	2 3
0	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
1	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
2	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
3	−	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
4	−	−	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
5	−	−	−	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−
6	−	−	−	−	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−
7	−	−	−	−	−	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−	−	−
8	−	−	−	−	−	−	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−	−
9	−	−	−	−	−	−	−	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−	−
10	−	−	−	−	−	−	−	−	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−
11	−	−	−	−	−	−	−	−	−	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−
12	+	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
13	−	+	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
14	−	−	+	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
15	−	−	−	+	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
16	−	−	−	−	+	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
17	−	−	−	−	−	+	−	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
18	+	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
19	−	+	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
20	−	−	+	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
21	−	−	−	+	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
22	−	−	−	−	+	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
23	−	−	−	−	−	+	+	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−

VERIFICATION O.K. !!!

PARALLEL PSEUDONOISE SEQUENCIES GENERATOR,

COMPUTATION OF THE TRANSITION MATRIX.(PROGRAM GPSA)
GEORGES ROGER L.D.M. 9/11/88

DEGREE OF CHARACTERISTIC POLYNOMIAL : 7

CHARACTERISTIC POLYNOMIAL : Z0 Z6 Z7

NB OF SIMULTANEOUS BITS: 32

```
RN31 : Z5  Z6
RN30 : Z4  Z5
RN29 : Z3  Z4
RN28 : Z2  Z3
RN27 : Z1  Z2
RN26 : Z0  Z1
RN25 : Z5  Z7
RN24 : Z4  Z6
RN23 : Z3  Z5
RN22 : Z2  Z4
RN21 : Z1  Z3
RN20 : Z0  Z2
RN19 : Z11 Z15
RN18 : Z10 Z14
RN17 : Z9  Z13
RN16 : Z8  Z12
RN15 : Z7  Z11
RN14 : Z6  Z10
RN13 : Z5  Z9
RN12 : Z4  Z8
RN11 : Z3  Z7
RN10 : Z2  Z6
RN09 : Z1  Z5
RN08 : Z0  Z4
RN07 : Z1  Z18
RN06 : Z0  Z17
RN05 : Z8  Z26
RN04 : Z7  Z25
RN03 : Z6  Z24
RN02 : Z5  Z23
RN01 : Z4  Z22
RN00 : Z3  Z21
```

PARALLEL PSEUDONOISE SEQUENCIES GENERATOR.

COMPUTATION OF THE TRANSITION MATRIX.(PROGRAM QPSA)
GEORGES ROGER L.D.M. 9/11/88

DEGREE OF CHARACTERISTIC POLYNOMIAL : 7

CHARACTERISTIC POLYNOMIAL : X0 X6 X7

NB OF SIMULTANEOUS BITS: 64

RN63 : X6 X6
RN62 : X4 X5
RN61 : X3 X4
RN60 : X2 X3
RN59 : X1 X2
RN58 : X0 X1
RN57 : X5 X7
RN56 : X4 X6
RN55 : X3 X5
RN54 : X2 X4
RN53 : X1 X3
RN52 : X0 X2
RN51 : X11 X18
RN50 : X10 X14
RN49 : X9 X13
RN48 : X8 X12
RN47 : X7 X11
RN46 : X6 X10
RN45 : X5 X9
RN44 : X4 X8
RN43 : X3 X7
RN42 : X2 X6
RN41 : X1 X5
RN40 : X0 X4
RN39 : X1 X18
RN38 : X0 X17
RN37 : X8 X26
RN36 : X7 X25
RN35 : X6 X24
RN34 : X5 X23
RN33 : X4 X22
RN32 : X3 X21
RN31 : X2 X20
RN30 : X1 X19
RN29 : X0 X18
RN28 : X13 X10
RN27 : X12 X17
RN26 : X11 X16
RN25 : X10 X15
RN24 : X9 X14
RN23 : X8 X13
RN22 : X7 X12
RN21 : X6 X11
RN20 : X5 X10
RN19 : X4 X9
RN18 : X3 X8
RN17 : X2 X7
RN16 : X1 X6
RN15 : X0 X5

RN13 : X9 X12
RN12 : X8 X11
RN11 : X7 X10
RN10 : X6 X9
RN09 : X5 X8
RN08 : X4 X7
RN07 : X3 X6
RN06 : X2 X5
RN05 : X1 X4
RN04 : X0 X3
RN03 : X15 X26
RN02 : X14 X25
RN01 : X13 X24
RN00 : X12 X23

W.J.Weber
W.E.Buell
May 9, 1988

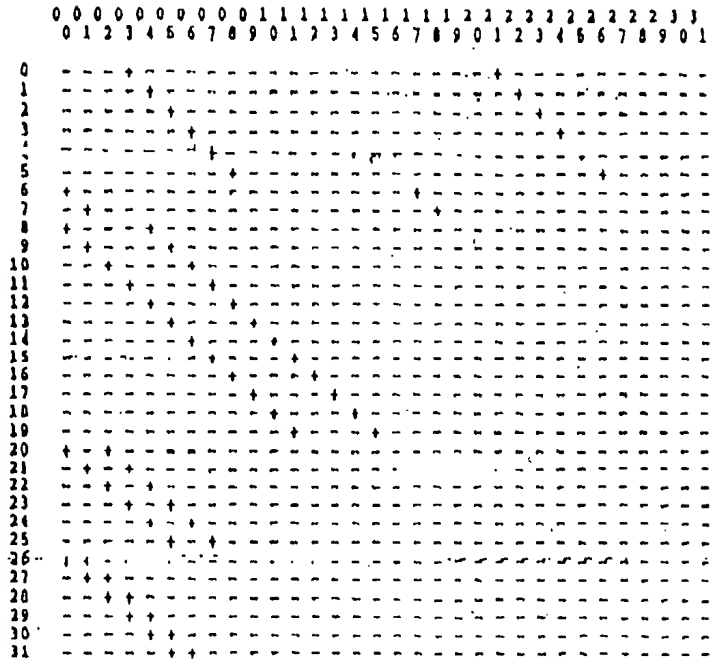
****
DEGREE OF CHARACTERISTIC POLYNOMIAL : 7

CHARACTERISTIC POLYNOMIAL : X0 X6 X7

NB OF SIMULTANEOUS BITS: 32

****

MATRIX:



VERIFICATION O.K. !!!

前記本発明の目的及び先の説明から明らかとなった目的は効果的に得られることは明らかであり、本発明の並列疑似乱数列発生機の範囲から離れずとも上記構成及び構造において所定の変更がなされ得るので、前記説明に含まれるまたは添付の図面に示された全ての事項は説明のためのものであって制限的ではないと解釈されたい。

更に、特許請求の範囲は、本明細書に記載の並列疑似乱数列発生機の一般的な及び特定の性質の全てを包含するものとし、本発明の範囲の全ての説明は、文字通りこのなかに含まれるものとする。

4．図面の簡単な説明

第1図は次に生成される値が多項式$1+X^m+X^p$によって定義されるようにP段シフトレジスタとして接続されているD型フリップフロップの使用を組み込んである直列疑似乱数列発生機を示す図、第2図は第5B図に示された直列PRGをエミュレートする8ビット並列PRGを示すブロック図、第3図はクロック信号を含む第2図に示された8ビット並列PRGの概略図、第4図は第1図に示された直列PRGの16ビット並列PRGの実現態様のブロック図、第5A図は、段nと段n-1との間の帰還関係を示す第5B図に示された直列疑似乱数列発生機のW個の出力を有する並列PRGの実現態様の概略図、第5B図は段P及び段P-1が段1の次の値を決定するために使用される帰還値である第1図に示されたものと同様の直列PRGの概略図、第6図は第5A図に示された並列PRGに対応する直列疑似乱数列発生機の並列実施態様の一般解のための遷移行列、第7図は多項式$1+X^5+X^7$のための出力(n)と出力(n+5)及び出力(n+7)の値との関係を示す図、並びに第8図は多項式$1+X^2+X^3+X^5$のための出力(n)と出力(n+2)、出力(n+5)及び出力(n+9)の値との関係を示す図である。

20…直列疑似乱数列発生機、24…並列疑似乱数列発生機、26…ラッチ、28,30,30'…排他的ORゲート。

出願人 アルカテル・エヌ・ベー
代理人 弁理士 川口義雄
代理人 弁理士 中村至
代理人 弁理士 船山武

図面の浄書(内容に変更なし)

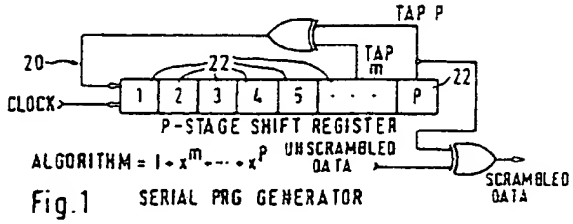


Fig.1 SERIAL PRG GENERATOR

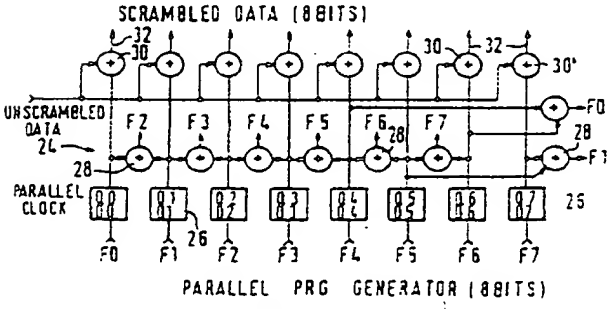


Fig.2

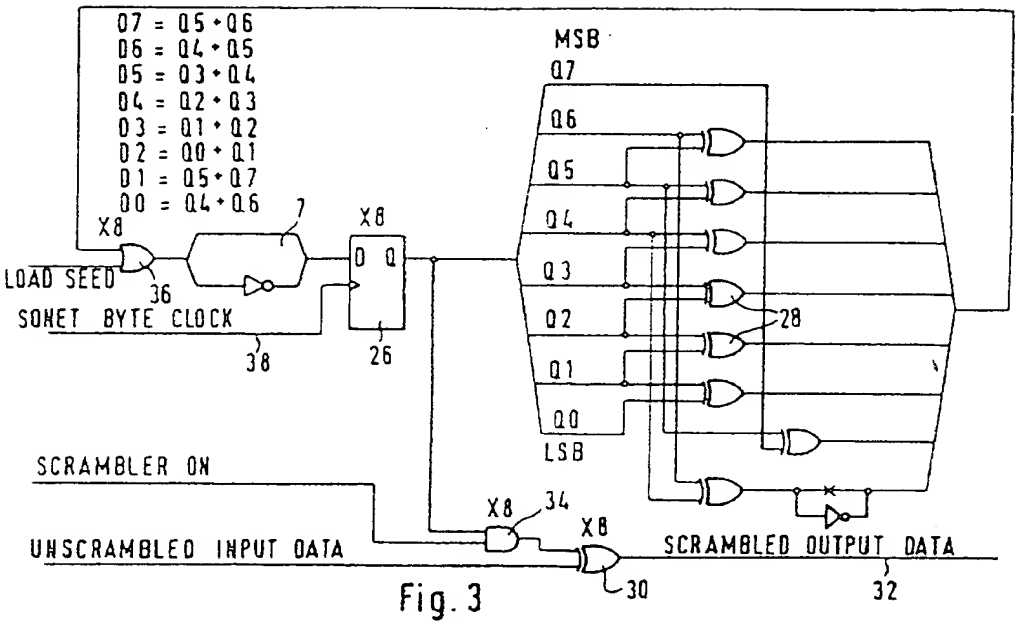

Q7 = Q5 + Q6
Q6 = Q4 + Q5
Q5 = Q3 + Q4
Q4 = Q2 + Q3
Q3 = Q1 + Q2
Q2 = Q0 + Q1
Q1 = Q5 + Q7
Q0 = Q4 + Q6
X8
LOAD SEED
36
SONET BYTE CLOCK
38
7
X8
D Q
26
MSB
Q7
Q6
Q5
Q4
Q3
Q2
Q1
Q0
LSB
28
SCRAMBLER ON
X8
34
X8
UNSCRAMBLED INPUT DATA
SCRAMBLED OUTPUT DATA
30
32

Fig. 3

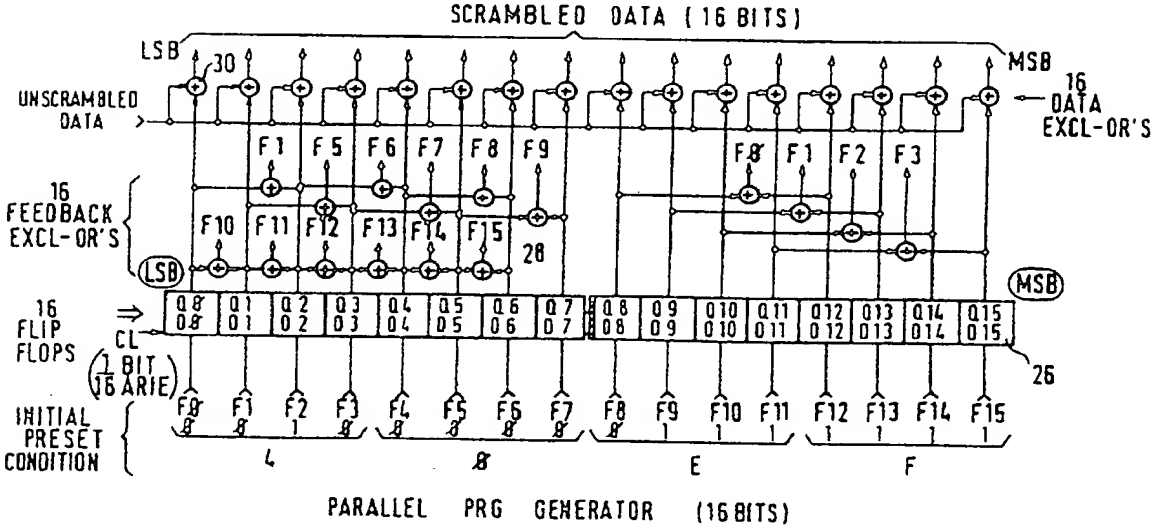
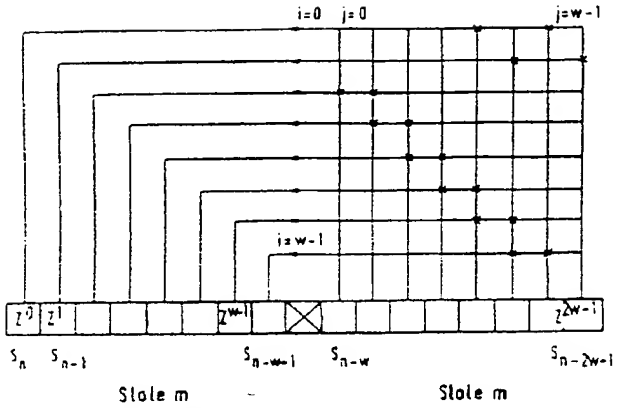
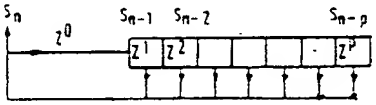

SCRAMBLED DATA (16 BITS)
LSB
MSB
30
UNSCRAMBLED DATA
16 DATA EXCL-OR'S
16 FEEDBACK EXCL-OR'S
F1 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15
F8 F1 F2 F3
28
16 FLIP FLOPS
CL
(1 BIT /16 ARIE)
Q0 Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8 Q9 Q10 Q11 Q12 Q13 Q14 Q15
26
INITIAL PRESET CONDITION
F8 F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15
4 8 E F
PARALLEL PRG GENERATOR (16 BITS)

Fig. 4



2) PARALLEL GENERATOR

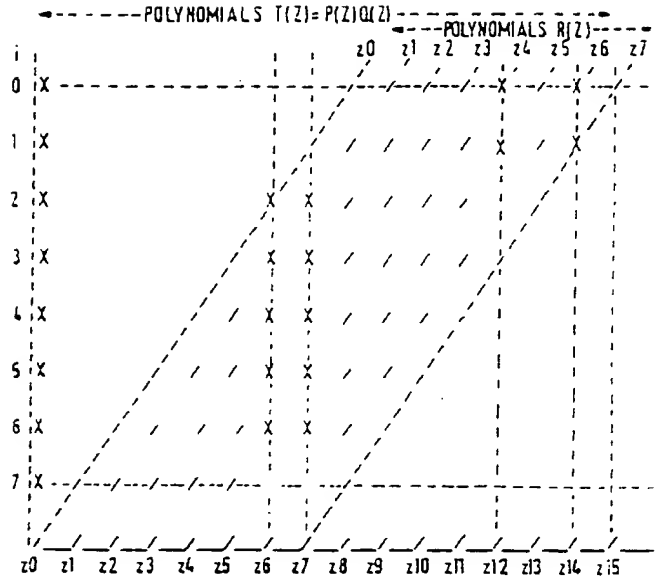
Fig. 5A



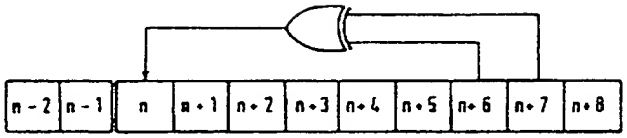
1) SHIFT REGISTER

Fig. 5B

Fig. 6



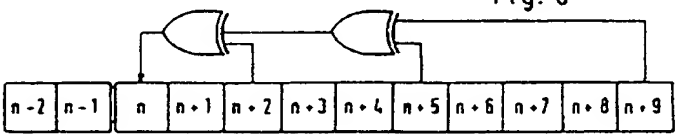
RELATIVE POSITION OF THE MATRIX ELEMENTS ('X')
AND OF POLYNOMIALS T(Z)=P(Z)Q(Z).
THE MATRIX IS IN THE PARALLELOGRAM



$Q(n) \equiv Q(n+6) + Q(n+7)$

Fig. 7

Fig. 8



$Q(n) \equiv Q(n+2) + Q(n+5) + Q(n+9)$

第1頁の続き

⑫発　明　者　ジョルジユ・アンドレ・シヤルル・ロジエ　フランス国、91240・サン・ミシエル・スユル・オルジユ、パルク・ドウ・ロルモイ、パビオン・マルグリット（番地なし）

手続補正書（方式）

平成2年9月4日

特許庁長官　植　松　敏　殿

1．事件の表示　平成2年特許願第122861号

2．発明の名称　直列疑似乱数列生成機をエミュレートするための並列疑似乱数列生成機及びその実行方法

3．補正をする者
事件との関係　特許出願人

名　称　アルカテル・エヌ・ベー

4．代　理　人　東京都新宿区新宿1丁目1番14号　山田ビル
（郵便番号160）電話（03）354－8623
（5200）　弁理士　川　口　義　雄
（ほか2名）

5．補正指令の日付　平成2年8月28日

6．補正の対象　図　面

7．補正の内容
(1) 黒色で鮮明に描いた適正な図面を別紙の通り補充する。
（内容に変更なし）

特許庁
2.9.5
出願課